CONTROL OF INITIALIZATION BIAS IN QUEUEING SIMULATIONS USING QUEUEING APPROXIMATIONS

DTIC
S ELECTE D
MAY 1 2 1995
F

A Thesis

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the Requirements for the Degree

Master of Science Systems Engineering

by

Patrick James Delaney

May, 1995

This document has been approved for public release and sale; its distribution is unlimited.

19950511 045

APPROVAL SHEET

This thesis is submitted in partial fulfillment of the requirements for the degree of

Master of Science Systems Engineering

Author

This thesis has been read and approved by the examining Committee:

Thesis Advisor

Accepted for the School of Engineering and Applied Science:

Accesion For	
NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

Dean, School of Engineering and Applied Science

May, 1995

ABSTRACT

A person often simulates a discrete event dynamic system (DEDS) with initial conditions that are not representative of the parameter he is trying to estimate. The reasons for this vary from no knowledge of the long-run state (steady state) of the modeled systems to complete disregard of this critically important piece of information. Regardless, the result of incorrect initial conditions in a simulation model is usually a biased output estimate. This bias based on incorrectly set initial conditions is commonly known as initialization bias. Many people have researched ways to control, detect, or negate this bias so we can derive accurate information from a simulation output sequence.

One way to minimize the initialization bias is to run a simulation for a large number of observations. Often this is impractical or not affordable in time and money. For example, a simple M/M/1 queueing model with a traffic intensity of 0.90 requires 13, 500, 000 observations to achieve an estimate with 1% absolute error to the true mean. This run length equates to 25 hours of real time. In the end, the estimate still has 1% bias to the true expected value. The question becomes, how much bias is acceptable?

The most widely used methodology of initialization bias control is that of data truncation. The idea is that the stochastic nature of the random variables will ultimately produce observations which are more representative of the steady-state of the system after a period of transience. The initial transient observations which bias an estimate are not representative of the steady state characteristics and can be deleted or thrown away. From that point in the output sequence to the end, the estimate will be less biased than taking the entire output sequence average. While these control methodologies are commonly known as heuristics, they do serve a purpose to provide the decision maker with the best possible information by doing the best they can with an output sequence.

This research focuses on producing a "good" estimate from sequentially correlated simulation output data. I evaluate the use of proven accurate queueing approximations to stochastically set the initial queue length from the approximated steady state distribution to derive a "better" estimate than empty and idle, without the use of pilot runs. I also evaluate how point approximations can assist in controlling the bias in output estimates of a desired performance parameter through four truncation heuristics. The end result is a less biased and more accurate estimator of the expected wait in a queueing model.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

The goal of my research is to develop a truncation method of a sequentially correlated simulation output sequence to provide a more accurate and more precise estimate of the steady state expected wait time in the queue as compared to the sample mean of the output sequence. To accomplish this I make use of apriori information of the system that a person creating a simulation model has. I incorporate an analytical approximation to assist me in getting my estimate. The following chapter introduces the problem of this research as well as reviewing the fundamental concepts in simulation modeling analysis.

1.1. Simulation

Simulation is a method in modeling analysis. Often, simulation is used for systems that are too complex and probabilistic for a simple analytical analysis. The system is modeled with prescribed input and the model output is used as information in a decision making process. We must understand what a simulation is and what a simulation produces before we can flatly accept output of a simulation experiment.

> "A simulation is the imitation of the operation of a real-world process or system over time. Whether done by hand or on a computer, simulation involves the generation of an artificial history of a system, and the observation of that artificial history to draw inferences concerning the operating characteristics of the real system."[1]

The fact that a simulation is only a model does not mean the performance of the model will be the actual performance of the real systems. In designing the simulation model, the designer makes several assumptions of the real world system. If these assumptions, such as the number of arrivals in an air passenger terminal in a set time, are erroneous, then no matter how the model is manipulated, the output will be just as erroneous.

Simulation analysis is second only to linear programming in frequency of use of operations research methods. Most simulation models focus on a discrete event dynamic system (DEDS). That is, "the modeling of a system as it evolves over time in which state

---

[1] Banks & Carson [2], p. 2.

variables change instantaneously at separate points in time."[2] There are several reasons to use simulation. As pointed out by Ravindran, Phillips and Solberg[[26], p. 376], some reasons to use simulation modeling analysis include:

> i) Through simulations, one can study the effects of certain information, organizational, and environmental changes of the operation of a system by making alterations in the model of the system and by observing the effects of these alterations on the system's behavior
>
> ii) A detailed observation of the system being simulated may lead to a better understanding of the system and to suggestions for improving it, which otherwise would be unobtainable.
>
> ii) Simulation of complex systems can yield valuable insight into which variables are more important than the others in the system and how these variables interact.
>
> iv) Simulation can be used to experiment with new situations about which we have little or no information, so as to prepare for what may happen.
>
> v) Simulation can serve as a "preservice test" to try out new policies and decision rules for operating a system, before running the risk of experimenting on the real system.
>
> vi) Simulation analysis can be performed to verify analytical solutions.

The advantages of simulation analysis are diminished when we consider how easy it is to use a simulation incorrectly. Simulation software makes it easy to create a model and obtain a performance parameter estimate from the simulation output sequence. If we accept a point estimate from even an extremely long simulation run, we run the risk of accepting an erroneous estimate as an accurate estimate of the true performance parameter's expected value. We must understand the underlying process. We must also concern ourselves with any method which may increase the variance of our analysis. I assume that the sample average of wait times in the queue converges to the true value. I focus my research on the initial transience of the output sequence. Additionally, since I ensure common random numbers (CRN) and synchronization in my experiments, I consider any

---

[2] Law & Kelton [20], p. 7.

variability of the estimates to be a result solely of the methodology and not from other random variables which act on the system.

1.2. Queueing Models

A queueing system is characterized by an arrival, a service, and if the server is busy at the time of arrival a waiting period in a queue. Queueing theory's applicability is prevalent throughout our society. A business application of queueing theory is a telephone communications network. Phone calls arrive on a particular line of communications only to find out the line is busy. There has been an enormous amount of research which focuses on minimizing the number of busy lines through network rerouting of phone calls. A toll free catalog order business wants the person who dials its number to get through immediately. If the line is busy, the caller may hang up and change his mind about ordering the product. This is lost business. We can easily see how the business wants to minimize the number of times a person calls and gets a busy signal. On the other hand simply placing an overabundance of operators who are barely busy costs the business in wages and benefits. Immediately, we can see the paradox of a queueing system. Often, businesses build simulation models and evaluate how the system performs. This information provides input in the decision making process.

The fact that a customer (phone call, person, etc.) has to wait in a queue is probabilistic in nature. This probability of a queue existing has an associated probability density function. We often simulate a queueing model because we have no knowledge of this probability distribution. Indeed, if we knew the distribution, there would be little need to simulate. We could get exact values of parameters we needed for our decision. Since we do not know this distribution, and analytically determining a solution for a particularly queueing system may be analytically intractable, we simulate.

1.3. Simulation of Queueing Systems & Data Analysis

The first step in simulation of a queueing system is to develop the model. A person creating the model, usually, has first and second moment information about the arrival and service times. This information, while it may not seem much, partially defines the underlying distribution of the queue characteristics. We need to consider all information we have when we develop the model. We also must understand we have no idea how the system will behave when we begin to simulate it. In other words, the initial conditions we start our model at may not be representative of the long run distribution. We must realize that this could bias our desired estimate.

To negate the effects of incorrectly set initial conditions in a simulation model, we would have to use an extremely long run to minimize the estimate bias of the performance parameter. The M/M/1 empty and idle queue is a good example of a lengthy transient period that requires a long run. As Whitt[37] points out, we would have to run an M/M/1 queueing model with a traffic intensity of 0.90, for 13,500,00 observation to obtain a parameter estimate with 1% absolute error to the true expected value. We know that a cumulative run mean of a sample path only approaches its true value asymptotically based on the law of large numbers. In a simulation, the output sequence observations go through several non-stationary distributions before they settle toward the steady state distribution. The early observations of the cumulative run mean are the result of individual random observations that may not be representative of the asymptotic stationary distribution. Accordingly, the distribution associated with each estimate observation may change. These changes in the distributions are non-stationary or ***transient*** distributions. The period it takes the cumulative mean to settle to one where the expected value is the true mean has become known as the ***warm-up period***. Once the cumulative mean begins to settle into an apparent stationary disposition, I consider the process to be in steady state. Theoretically, steady state is the asymptotic state of the system. Since we cannot truly attain an infinite simulation run, I consider ***steady-state*** to be that point in which the simulation output sequence appears to be a covariance stationary process. I use the definition for covariance stationary from Law & Kelton [20].

$\mu_i = \mu$ for $i = 1,2,.....$ and $-\infty < \mu < \infty$

$\sigma_i^2 = \sigma^2$ for $i = 1,2,.....$ and $\sigma^2 < \mu < \infty$

and $C_{i,i+j} = \mathrm{Cov}(X_i, X_{i+j})$ is independent of i for $j = 1,2,.....$

The transient period is one major reason for bias in the simulation end estimate. The transient period is a direct result of the simulation beginning in conditions that are not representative of the true stationary asymptotic distribution. The major problem is we have no idea how long a transient period will be. The transient period may vary significantly among sample paths. We know by the Law of Large Numbers that we can get an estimate closer to the true expected value the longer we run a simulation model. Since this is computationally inefficient, we must consider how to get the most precise and accurate estimate from the data we have. Consider the bias of the estimate resulting from incorrect initial conditions to be ***initialization bias***. I concede that with a finite run length there will exist some bias of the estimate, however, I believe I can control the bias of our estimator regardless of the initial conditions. Researchers have developed several methods in attempting to control the bias based on the initial transience.

One method to minimize or eliminate initialization bias, is to set the initial conditions to those representative of the stationary long run distribution of the system, before a run. This is often impossible since the stationary distribution is unknown. If I could select an initial condition from the steady state distribution of the desired performance parameter, then there would be little need to simulate.

I can set the initial conditions of a system either deterministically or stochastically. Kelton[17] explores both methods. Deterministically setting the initial conditions sets a constant value such as the mode or value near the mean and executing the simulation with these set initial conditions. This should shorten the transient period and ultimately provide a better estimate than empty and idle initial conditions.

The problem with setting the initial conditions deterministically is it ignores other possible states of the system. Kelton demonstrates how using geometric distributions to stochastically set the initial conditions provides a more representative analysis of the system. Whitt [40] develops approximations for the steady state distributions of the number in the queue and system. Using the approximations to stochastically set the initial conditions should give us an end estimate closer to the true expected value than an empty and idle system.

Two methods to produce independent observations of the value we are trying to estimate are Batch Means and Replication/deletion. (See Chapter 3.) The batch means method provides us with a way to derive the variance of our estimator, using only one long run to produce the point estimate. Note that the average of the batch means is our estimate of the true expected value. This is also the cumulative run mean. I concede that a longer run has a higher probability of going through the transient period than independent replications. However, we have no idea how long a transient period is. One long run may result in a more biased estimate than a point estimate across independent replications. For this reason, I endorse setting the initial conditions across independent replications. This allows us to control the probabilistic nature of what the system sees as its initial condition and allows for more accurate and precise statistical analysis of the output data.

Another method to control the initialization bias is to truncate a portion of the sample data. The remaining data provides our point estimate. A finite computational budget may never get us through a transience phase. This can ultimately result in an estimate that is neither accurate nor precise; even with truncation. Additionally, when we truncate an observation from a sample data set, we completely change the characteristics of the sample mean. We could conceivably change the distribution as well as the expected value of the data set after a truncation. For this reason, we must consider what the ultimate goal of a simulation is. I consider that accurate information for a decision making process is the ultimate end purpose of a simulation. For this reason, I consider anything short of a precise and accurate point estimate to be not very useful.

We can attempt to manipulate the data to be more representative of steady state by eliminating transient observations in a sample data set. Several researchers have developed methods to truncate initial transient observations. The goal of all heuristic methods is to obtain a more precise and accurate long run performance estimate. Heuristics by Fishman [6], Welch [35], Conway [5], Gafarian [7], Kelton & Law [15], White [36] and others differ in their approach to control the initialization bias, but all incorporate some form of truncation. Most truncation heuristics require **pilot runs** to determine the characteristics of the output data and an estimate of the truncation point. The Welch [35] Plot allows us to visualize the transient period across pilot runs and obtain a "guesstimate" of the average warm-up period, based upon these pilot runs. The cumulative run means averaged across pilot runs gives the simulator a visual clue of where the apparent covariance stationary phase begins. At that point, the simulator would clear his statistic (or truncate) and begin a new estimate. If these pilot runs were anomalies, then we could possibly choose an incorrect truncation point. The Welch Plot does allow us to see when the cumulative run mean appears to settle into a covariance stationary process. We must not lose sight of the fact that this is a plot across replications so it is merely an estimate of the average warm-up period. Each simulation sample path has its own unique transient period. We must consider this in our analysis. Pilot run heuristics do allow us to gain insight to the system and then use this knowledge to run our experiments. Nevertheless, the waste of computational budget time is a shortcoming of this method.

1.4. Overview of the Methodology

Fishman [6] discusses the penalty of increased sample variance we may incur from truncation, even though we may see a decrease in the initialization bias of a simulation estimate as a result of truncated data. He recommends using the mean square error as an examination of this truncation penalty. MSE $= (X - \theta)^2 + \sigma^2$, where X is an estimator of the true expected value, θ, and σ^2 is the variance of the estimator. Fishman used the sample mean as the true mean in his calculation of the mean square error (MSE);

MSE $= \left(X_i - \overline{X}\right)^2 + \hat{\sigma}^2$, where X_i is the observation of the cumulative mean, $\overline{X}$ is the sample mean, and $\hat{\sigma}^2$ is an estimate of the variance of the estimator. I consider using the Whitt [40] approximation for the steady state expected wait time in the queue as the true mean and bootstrap off Fishman's MSE penalty examination.

Modifying on Kelton's[17] initial conditions study and Fishman's MSE idea, I provide new methods of point estimation of waiting time in the queue using proven accurate approximations to assist in the estimation process to stochastically set the initial queue length and then truncate the output data at the point where the minimum estimated MSE occurs, thereby minimizing the truncation penalty. I then calculate an estimate from the remaining data.

I used the Extend®, simulation package on a Macintosh Quadra 840AV for this research.

1.5. Purpose of this Research

Several significant reasons which support this research include:

i) If I can put together a process for queueing systems which produces a good approximation of steady state parameters prior to a simulation run, I can drastically reduce the time used to develop and construct models, execute the simulations and collect data. This could ultimately equate to monetary savings in business.

ii) Using design criteria of the system to give us first and second moment information of the proposed system, I can intelligently use this information to set initial conditions from the approximate steady state distribution . This will ultimately allow us to eliminate pilot runs.

iii) If I can successfully control the initialization bias and variance through minimizing the mean square error, I can automatically produce the best possible estimate from each sample path.

iv) Portable code allows the algorithms herein to be adapted by any simulation package or any other research student.

v) This method will ultimately provide a decision maker with more precise and accurate information.

vi) This research presents several new truncation rules to use in simulation modeling.

1.6. Organization of Thesis

Chapter 2. is a review of literature devoted to initialization bias in simulations as well as queueing approximations. Chapter 2 provides the theoretical, but more so, the technical background and explanation of the significance of the initialization bias problem. Chapter 3 is the thesis methodology. It describes setting the initial conditions; stochastically and deterministically. It provides the reader with a detailed explanation of the heuristics that are used in the experiments and the design of the experiments themselves. Chapter 4 is the results and analysis of the experiments described in Chapter 3. I perform batch means and replication/deletion analysis. Chapter 5 is a review and evaluation of my research and a discussion of possible future research topics. While I have not included the ModL® code I wrote for this thesis, I have included the algorithms in Appendix A. This will allow anyone who desires to follow on in this research topic the flexibility to use any computer language. Additionally, the actual Extend® models are available if one desires. Appendix B is the results of my experiments in detail. Appendix C is a glossary of some of the terms I use in the thesis. They are presented to clarify any confusion that may arise. The thesis concludes with an extensive reference list at Appendix D.

CHAPTER 2. REVIEW OF RELATED LITERATURE

Initialization bias has received a great deal of attention in the simulation community. Techniques or heuristics have been developed which attempt to minimize the statistical bias associated with the initial transient phase. This chapter introduces the reader to the problem of initialization bias. It then reviews ways to set initial conditions prior to a simulation run, test for initialization bias after a simulation run, and reviews heuristics for dealing with the bias through the truncation (deletion) of the transient phase biased data. Finally, there is a discussion of queueing approximations and how we can use the approximations in a point estimate process.

I categorize the topic of initialization bias into three categories: i) setting initial conditions, ii) test for initialization bias in simulation output, and iii) controlling initialization bias. This chapter presents a literature review of the research in these categories. While there is ongoing research in controlling initialization bias using other methods (time series methods and others), I focus on control of initialization bias through truncation heuristics.

2.1. Setting the Initial Conditions in Simulations

In an attempt to minimize the bias associated with a nonstationary process, we can set initial conditions which will reduce the transient phase and thereby reduce the bias. Intuitively, one of the first questions we ask when we build a queueing simulation model is, "Should the state variables of the simulation be started at 'empty and idle' or front loaded to an initial value?" If the answer is to front load, the system, then immediately we would need to know at what value(s).

Initial conditions can be set in two ways: deterministically or stochastically. While almost every reference used in this review touches on the topic of setting initial conditions, Kelton [17] focuses his paper on this topic alone. He explores the best stochastic and deterministic ways to set initial conditions in replicated simulation runs. Kelton uses replications rather than one long simulation run with batch means because the method is extremely simple and replications produce independent, identically distributed observations of a performance parameter. The one shortcoming of replications is each run has to go through the initial transient phase; however, Kelton believed that the advantages of replications outweighed the shortcomings.

Setting initial conditions in a deterministic manner refers to setting each replication in an identical state. Wilson and Pritsker [42] determined that setting the initial conditions at a value near the mode of the steady state distribution reduced the transient phase. Kelton and Law [15] determined in several queueing models that setting the initial conditions at a value close to the steady state mean "induced comparatively short transients." The problem with both approaches is that some prior knowledge of the mode or mean would have to be available. This at the very least would require a single pilot run, with no guarantee that the output would be the correct value.

If it were possible to set the initial conditions from the steady state distribution, then there would be no transient phase. This is nearly impossible and extremely improbable. Accordingly, Kelton [17] explores the use of maximum entropy to determine a probability distribution to select initial conditions from. The two distributions Kelton ascertains through the maximum entropy process are a Uniform distribution from 0 to "m", and a Geometric distribution with parameter p; where $p = 1/(\nu+1)$ and ν is a "guess" value of the mean. Kelton points out, that you must make some educated arbitrary selection of input to the system (m or ν), in using the Uniform and Geometric distributions. However, this educated arbitrary selection is now a value of a parameter of an initial distribution, rather than a deterministic value of the steady state. If we select a deterministic value incorrectly, then all replications will be off according to this value. However, the parameter associated with a distribution allows for a better probability of the correct initial condition to be established which in turn reduces the error. As in the deterministically set initial conditions, one would use pilot runs to get a better idea of the parameter values.

In evaluating deterministic versus stochastic initial conditions, Kelton used three criteria: i) Plots of expected transient response (or an estimate of) as a function of discrete time; ii) The percent bias of the expected value of an estimate with the steady state mean; and iii) The time required to attain steady state. Kelton performed this process on three queueing models (M/M/3, M/E_4/1, E_4/M/1), a time sharing computer system and a manufacturing model, to show the applicability for a more complex system. In each system, running a pilot run and then setting the initial conditions using the Uniform or Geometric distribution methodology had a better reduction in the bias of an estimate than using a deterministic initial condition. In fact, in the manufacturing model, the bias associated with the deterministic approach had a bias of 39% while using the uniform distribution to set the initial conditions per replications reduced the bias by 60%. Kelton does concede that deterministically setting the initial conditions to an *optimal* value is better than a stochastic method. In Kelton and Law [15], the authors show that the *optimal* deterministic value in queueing simulations is larger than the mean and exceeds the mode. They also suggest that determining an estimate of the mean and initializing at a value equal to or greater than this mean value is better than the mode. One must note, however, that it is nearly impossible to determine the optimal value, and the stochastic method is a safer way of initialization.

Currently, there is research on how to approximate deterministic values as well as stochastic distributions. There is the hope that using an approximated stochastic process to set initial conditions in a simulation will drastically reduce the transient phase and ultimately produce a "good" statistic.

2.2. Truncation Heuristics in Output Analysis

While there is ongoing research using time series methodologies and other ways to control initialization bias, the most widely used is some form of deletion or truncation rule. Truncation rules are heuristics which attempt to delete a portion of the simulation run where the transient bias is most severe. This normally occurs at the beginning of the simulation run. Going from the transient phase and reaching the steady state level of the run is known as a warm up period. The truncation heuristics attempt to determine at what point "steady state" occurs. At that point, all previous observations are discarded and the remaining data is used in the estimate.

In 1978, a comprehensive paper by Gafarian, et. al [7] evaluated the most commonly used initialization truncation heuristics. These truncation heuristics are known as; i) The Conway Rule (R.W. Conway); ii) The Modified Conway Rule; iii) Crossing of the Mean Rule (G. S. Fishman); iv) Cumulative Mean Rule (G. Gordon); and v) The Gordon Rule (G. Gordon). The conclusion of this paper was that all these heuristics performed badly. However, the Gafarian study used a precision of zero tolerance when estimating the expected value (in this case queue wait time). Intuitively, one can see that the problem with this analysis is that the only way the precision of an estimate is approximately zero is if the simulation run length approaches infinity. Had Gafarian used a confidence interval to test for a desired precision half-width, they would have seen that the heuristics do perform a service in simulation modeling. The evaluation criteria that Gafarian, et al used in their experimentation were: i) Accuracy (The ratio of the expected value of the estimate and the true value. If the value is close to 1 then the heuristic is accurate); ii) Precision (A measure of the variation. If the precision measure is close to 0, then the heuristic is precise); iii) Generality (The heuristic works well across a variety of systems); iv) Cost (Expense in computer run time the heuristic takes); and v) Simplicity (A heuristic which is easily used by the average user of large scale simulations).

Similarly, Wilson and Pritsker [41] [42] review and evaluate the above mentioned heuristics. They used the same criteria as Gafarian, et. al. to evaluate the heuristics, but in their research, Wilson and Pritsker applied an initial condition rule first and then evaluated the heuristics. Where Gafarian, et. al. did not consider the use of confidence intervals, Wilson and Pritsker do. They estimate the MSE and develop a confidence interval around the estimator. Wilson and Pritsker found in many of their experiments that the truncation heuristics did in fact reduce the bias associated with the initialization. However, the MSE increased. This meant that the variance increase was indeed more significant than the bias reduction. As I stated earlier, they determined the mode was the best deterministic setting of the initial conditions in a simulation to reduce the transient period and thereby control the initialization bias.

Other researchers have published papers outlining their own heuristics. They include Law & Kelton's [21] Regression Based Algorithm, Snell and Schruben's [31] Weighting Simulation Data and White & Minnox's [36] Confidence Maximization Procedure, and Glynn & Heidelberger's [9] control of initialization bias in parallel simulations.

The following pages give a brief synopsis of each truncation heuristic and, hopefully, provide the reader with thought provoking topics. We must remember that each rule is indeed a heuristic, the ultimate goal is to acquire a better estimate of a desired performance parameter without the expensive computational cost of an extremely large run length.

Heuristic	Conway Rule (Conway 1963)
Basic Process	Truncate a series of observations until the first of the series is neither the maximum nor the minimum of the remaining data set. This point is the truncation point.
Pros	Extremely simple to apply to an output sequence.
Cons	Underestimates the truncation point badly for most cases (Wilson & Pritsker)
Comments	A run length of 5 is highly improbable to reach steady state. The heuristic is an attempt to determine what point to delete data to get a better estimate. A run length of 5 is hardly enough to produce a possible point. As such only 1 out of 100 runs with run length = 5 was successful in detecting initialization bias.
Heuristic	Modified Conway Rule (Gafarian, et. al., 1978)
Basic Process	Continually look backwards from the simulation output to find the first observation that is neither a maximum nor a minimum of all the previous observations. This point becomes the truncation point.
Pros	The run continues until the criterion is satisfied. No run length needs to be predetermined.
Cons	1) There exist the possibility that every observation is deleted except for the last observation. At this point the estimate may be good, but the variability will be significantly large. 2) Similarly, there exist the possibility that no data is truncated. This does not mean that no bias exists. Therefore your estimate is still biased to some degree.
Comments	This process was modified by Gafarian et al. The evaluation was once again skewed with a self-fulfilling prophecy of the inadequacy of the heuristic. In evaluating this heuristic, Gafarian, et al replicate a maximum of 10 times. Statistically, if one were attempting to get a good estimate of a mean value using replications, then one would have to use a minimum of 30 replications. It would be interesting to see their evaluation with more replications.
Heuristic	Crossing of the Means Rule (Fishman, 1973)
Basic Process	Looking backwards at the observations, comparing a cumulative mean to the observations, the truncation point becomes the designated number of allowable crossings of the cumulative mean determined prior to the simulation run. The greater the value allowed for crossing the mean, the more likely the initialization bias will have been resolved by that point.
Pros	Produces a good estimate of the desired statistic.
Cons	1) Extremely conservative rule which unnecessarily deletes data. This results in higher variability. The precision of the M/M/1 model with $\rho \geq 0.9$ is closer to 1. Closer to zero the better. 2) It overestimated the truncation point in most cases where the traffic intensity, ρ, was low.
Comments	A conservative rule such as this may in fact eliminate bias but have a variance that is large. The mean square error may also increase because the reduction in bias is offset by a larger increase in variance.

Table 2.1. Synopsis of Related Truncation Heuristics

Heuristic	Cumulative Mean Rule (Gordon, 1969)
Basic Process	Prior to a simulation run, determine the number of replications and the run length, all of which remain constant. Additionally predetermine the initial conditions. Plot the cumulative mean over the plot of each run and select the observation which appears to be stable.
Pros	Mathematically tractable and easily implementable
Cons	Large truncation points involved excessive computing cost.
Comments	1) Extremely sensitive to the number of runs and the run length. The more runs, the closer the cumulative mean approaches the true mean. The shorter run lengths do not mask the transient phase as easily as a longer run length.
Heuristic	Gordon Rule (Gordon, 1969)
Basic Process	Similar to the Cumulative Mean Rule, the Gordon Rule plots the estimate of the sample standard deviation versus the number of observations, n, on log-log paper and selecting as the truncation point that point where the plot steadies to a straight line.
Pros	Visualization is easiest form of heuristic. A program can be written to automatically plot the Log-Log plot.
Cons	Large truncation points involved excessive computing cost.
Heuristic	Moving Average Rule (Welch, 1983)
Basic Process	Using a visual representation of the output, one can use a moving average to calculate the average of the most recent "k" observations at each data point. The user selects the size "k". This smoothes the fluctuations in the output response and visually depicts the onset of steady state: The truncation point.
Pros	Applicable to all models, not just small well behave simulation models.
Cons	Basically reduces the sample size by n/k
Comments	The Welch Plots are the most widely used visualization of steady state and truncation selection. The problem is as with most heuristics, you have to run a pilot run of the simulation to ascertain the characteristics of the output.
Heuristic	Regression Based Algorithm (Law & Kelton, 1983)
Basic Process	Make several independent replications and collect the estimate over each run into "m" observations. Batch these observations and compute a batch mean. Fit a straight line through the batch means and check for zero slope. At that point, steady state has occurred and it becomes your truncation point.
Pros	1) Mathematically tractable and easily understood. 2) Performed well for a variety of queueing models.
Cons	Basically only good for monotonic processes. If one does not know the stochastic characteristics of the process, then using this method might be in error. Additionally, it requires several pilot runs to get an estimate of the truncation point and run length.

Table 2.1. Synopsis of Related Truncation Heuristics (continued)

Heuristic	Weighting Simulation Data
Basic Process	Using a least squares method (ordinary or generalized),
Pros	Snell and Schruben's experiments compare the least squares result to that of the analytically derived Conditional Means Square Error. (Unlike Fishman, 1972, who proposed minimizing the run mean square error) The Mean Square Error focuses not only on the bias, but also on the variance. By minimizing bias, one does not guarantee a good estimate of a performance parameter. Schruben and Snell attempt to find optimal point where the conditional mean square error is minimized.
Cons	1) Not as mathematically tractable as Law & Kelton's Regression Heuristic. 2) Calculating the Mean Square Error requires estimating both the bias and the variance of the sample statistic.
Heuristic	Confidence Maximization Rule (White & Minnox, 1992)
Basic Process	Minimize the half width confidence interval of the observed sample mean. The point of the simulation output that does this is the truncation point.
Pros	1) Sound basis for the statistical analysis. 2) Easily computed and possible to attach to simulation language.
Cons	1) Tests performed were only on queueing models 2) Assumption that at the onset of steady state, the confidence level peaks is yet to be proven.
Comments	White and Minnox discuss how many heuristics have a limited scope, yet their experimentation of their rule is only on queueing models, however the sound basis for the minimization of the confidence interval half-width makes it intriguing to test the rule on more complex systems.

Table 2.1. Synopsis of Related Truncation Heuristics (continued)

2.3. Testing for Bias

If a system's initial conditions are representative of the steady state conditions, then it is possible that no bias is present. This is rarely the case. It is necessary to have a method to see if any bias exists in the output of a simulation model.

Schruben [30], published a methodology for detecting initialization bias in simulation output. He developed a method using the theoretical applications of **φ—mixing** (phi-mixing) and **Brownian Bridge**. These are concepts that, although grounded in sound probability theory, are difficult to understand and implement for the average simulator. The idea is that the simulation output is basically a continuous time stochastic process. The unknown function of a mean performance parameter represents the possible shift in the output mean. Schruben applies the theory of dependent stochastic processes assuming that the random variable X_i is stationary and **φ—mixing**(phi-mixing) with a finite variance. This allows Schruben to use the central limit theorem to obtain a limiting distribution of the desired test statistic. With these assumptions, Schruben uses a Brownian Bridge as a limiting stochastic process on the interval starting at 0 and returning to 0. Schruben points out that the Brownian bridge has four significant properties; the characteristics of which converge asymptotically the partial sums of deviations about the mean of the Brownian Bridge process. From this basis, Schruben develops the standardized test sequence $T_n(t) = [tn]S_n([tn])/(\sqrt{n}\sigma)$; $t \in (0,1]$, with $T_n(0) = 0$.

Schruben's test procedure basically evaluates whether or not a sample estimate would be unusual if the output contained no initialization bias. The idea is that a large positive maximum value of the scaled test sequence $T_n(t)$ is unusual if no negative initialization bias is present. Schruben's next step is to develop the equations to estimate the probability that a test statistic, $T_n(i)$, is more unusual than the observed statistic. If this is the case, then the associated probability (denoted as $\hat{\alpha}$ by Schruben) is larger than the probability of no unusual test statistic. Smaller values of $\hat{\alpha}$ indicate that the resulting simulation output is highly unusual for an unbiased run. The test procedure step by step is simply:

1) Find $\hat{s}$, $\hat{s} = \sigma T_n(\hat{t})$, where $\hat{t}$ is the time of the observed first maximum.
2) Compute $\hat{\alpha}^2$ and ν (estimate of the variance/degrees of freedom)
3) Set $\hat{t} = k/n$ and compute $\hat{h}$ (hypothesis test)
4) Compute $\hat{\alpha}$
5) Reject a hypothesis of no negative bias if $\hat{\alpha} < \alpha$

Schruben experiments his test for initialization bias on five complex systems. In all experiments the test worked well. While there is always the possibility of Type I and Type II errors in rejecting or accepting a hypothesis, the test's performance was such that this should not be a major concern. Schruben does point out that if Type I or Type II error are a major concern, one must decide to increase the run length or truncate some data. These areas are another topic in the search for control of initialization bias.

Expanding on his earlier work, in 1983 Schruben and Singh [29] present an optimal test for initialization bias in simulation output. While this test was created as an attempt to control bias, it has predominantly come to simply be used as a test for initialization bias in simulation output after one uses a truncation heuristic. In fact, truncation of some data is an assumption Schruben uses to show the power of the optimal test. The bias test uses statistical theory (likelihood ratios) to derive the optimal form. The test is simply whether or not the observed output value is consistent with the hypothesis that the mean does not change throughout the simulation run. Rejecting this hypothesis means there is bias present in the output.

Using the Neyman—Pearson Lemma, Schruben & Singh derive the likelihood ratio for the hypothesis. Using the concepts derived in his 1982 paper concerning the sensitivity of the cumulative sums, Schruben & Singh point out that the sum is highly sensitive to nonstationary means. Using a weighted average, Schruben and Singh develop the optimal test to be: $T = \sum_{k=1}^{n} c_k k S_k$ with $c_k = a_k - a_{k+1}$. That is, the test statistic, T, is the weighted average optimal test statistic, S_k is the cumulative sum process $S_k = \overline{Y}_n - \overline{Y}_k$ *where* $\overline{Y}_k = 1/k \sum_{i=1}^{k} Y_i$, and a_i are the selected weights which reflect the changes in the output mean due to initial conditions. If $a_i = 0$, then no bias is present in the output. Since in an actual application the "a's" are not known, one cannot determine the optimal value for a. One does know that in a stationary process, $c_k = 0$. Additionally, in simulations such as queueing systems, one can suspect the behavior of the transient mean (positive or negative bias) and select a weight accordingly. Schruben uses $c_k = 1 - (k/n)$ in his experiments, which is optimal for a quadratic mean function. While this test statistic is sound, it does not consider the variability of the stochastic process. Additionally, Schruben's test can lead to erroneous inferences based upon its results. If the finite sample data set appears to settle into a stationary process because of its run length, then the Schruben test may show, after truncation, that the estimate has no initialization bias when in fact it could be a significantly biased estimate.

Schruben and Singh describe a method to estimate the variance of the T statistic: $\mathrm{Var}(T) = n^3\sigma^2/45$. Since σ^2 is unknown, we must estimate it from the simulation output. While Schruben and Singh do not perform the estimate of the variance, they do propose the use of either the autoregressive process to estimate the variance or the batch means process on the output. Using the estimate of the variance and the weighting test described above, Schruben and Singh derive an estimate of T which is reasonable and performed well in all tests. The final estimate of the test statistic for initialization bias is: $\hat{T} = \left(\sqrt{45}/n^{3/2}\,\hat{\sigma}\right)\sum_{k=1}^{n}(1-k/n)k\left(\bar{Y}_n - \bar{Y}_k\right)$. The step by step process using this test for initialization bias is:

1) Compute $\hat{\sigma}$ and the degrees of freedom using either batch means or autoregressive methods

2) Compute the test statistic $\hat{T}$.

3) Using the t-distribution tables, reject the hypothesis that the mean does not change if $\hat{T} > t\,(d, \alpha)$, where d is the degrees of freedom and α is the desired confidence level. (i.e., $\alpha = 0.05$ for confidence level 95%).

We can see that this optimal test for initialization bias is more readily understandable. In fact, it has become the norm for testing for initialization bias in simulation output.

Another test for initialization bias was presented in 1989 by Vassilacopolous [34]. Vassilacopolous' test is based upon ranking the simulation observations. Like Schruben [30], Vassilacopolous assumes the output process in the simulation is stationary and φ—**mixing** with finite variance. Vassilacopolous ranks tied observations with the same value for an equal rank. The rank definitions of the observation x_k with N observations is:

$$R_k = \sum_{i=1}^{N} S(x_k - x_i) \text{ where } S(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases}.$$

Vassilacopolous next develops a new process defined as: $U(k) = 2W_k - k(N+1)$; $k = 1, 2, \ldots, N$, where $W_k = \sum_{j=1}^{k} R_j$ *and* $U_N(0) = 0$. The test for initialization bias is $C_N = \max_{1 \le k \le N} |U_N(k)|$. This is the two-sided test for initialization bias. Vassilacopolous also develops the test for positive and negative initialization bias. Once again, like Schruben [30], Vassilacopolous uses the Brownian bridge to determine the approximate size of the critical regions of the test from the null hypothesis probability distribution. Vassilacopolous derives the limiting distribution for the test to be $\max_{0 \le t \le 1} \{|B_t|\} = (3/N(N+1))^{1/2} C_N$. This two sided limiting distribution turns out to be the same as the two-sided Kolmogrov-Smirnov goodness of fit test statistic. This means that C_N is already determined. The one sided tests are also known. From this, Vassilacopolous shows the significance probabilities associated with C_N. Vassilacopolous' step by step process for initialization bias testing is:

1) Find the ranks of the observations R_k and calculate $U_N(k)$
2) Find c = max $|U_N(k)|$ and significance probability, $\hat{\alpha}$,associated with it.
3) Reject the hypothesis that no initialization bias is present if $\hat{\alpha} < \alpha$

Unlike Schruben's extensive testing on complex systems, Vassilacopolous performs experimentation of his test on the simple M/M/1 and M/M/4 queueing models. This questions the power of Vassilacopolous' test. Ma and Kochar [22] compare Vassilacopolous [34] and Schruben and Singh [29]. One significant advantage of Vassilacopolous' Rank Test is one does not have to estimate the variance of the stochastic process. For this reason and its simplicity, Ma and Kochar believe that Vassilacopolous' test is more likely to be used by the simulation community. In comparing actual implementation of the tests, Ma and Kochar experimented with each test using simulation runs with no truncation, but biased data inserted, and simulation runs with truncation and biased data inserted. They found that both tests performed satisfactorily. Schruben's optimal test was able to determine more biased runs than the rank test. Once again, however, one must consider the ease of implementation and use of the rank test. Even though the rank test does have some shortcomings, its advantages make it the one which Ma and Kochar recommend. One thing that Ma and Kochar note affects both tests is that if the variance of the stochastic process is larger than the size of the initialization bias it is difficult to detect the initialization bias. To that end, they suggest that the variance be reduced using batch means or replications.

Since I am considering a finite run length, I know the estimate will not achieve its asymptotic true expected value. Accordingly, I concern myself with analyzing stochastically setting the initial conditions and evaluating the performance of the truncation heuristics to provide a more accurate and precise point estimate of the true expected value of the performance parameter. I provide the tests for initialization bias discussion as one which is related to my research. Additionally, the results of using approximation assisted point estimation may have applicability in future research topics in testing a data set for initialization bias as well.

2.4. Queueing Approximations

A queueing approximation allows us to perform a quick analysis of a system without a simulation. Researchers have developed several queueing approximations that allow us to perform this quick analysis. Tijms [32] devotes an entire chapter to this topic. There are three basic methods for approximating the performance of a Queueing system. (Gross [11]) They are process, bound and inequalities, and system approximations.

A process approximation is where "the actual problem is replaced by a non-queueing one which is simpler to work with. The primary examples of interest are the use of continuous-time diffusion models to solve queueing problems and the use of creative probability arguments on random walks, stochastic convergence, and the like to solve heavy-traffic and nonstationary problems."[3]

A bounds and inequality approximation is one that attempts to take advantage of the limiting values of a more accurate system to approximate the value of a system that does not have the bound or inequalities needed to solve analytically. The approximation tries to "multiply the bound by a fractional function in ρ that itself approached one with ρ."[4]

A system approximation attempts to use known systems to approximate the performance of other systems. An example is using the $M/E_k/m$ model to approximate an M/G/m system.

I focus my research on system approximations. In particular, I focus on two-moment approximations for the waiting time in the queue. Tijms [32] illustrates the two-moment system approximation for the M/G/m Queueing model. Whitt [40] develops two-moment system approximations for the GI/G/m Queueing system using the analytically tractable M/M/m Queueing system. Whitt's approximation allows us to easily calculate the expected performance in a general queueing system. Whitt shows his approximations to be accurate with average absolute errors of 10% to the true expected value for cases considered.

I assume that a person designing a simulation model will have first and second central moment information of the system arrival and service times. This partially defined distribution is enough for us to use Whitt's approximations in my heuristics. Tijms [32, p.295] does note, however, that the accuracy of an approximation degenerates as ρ decreases. This is also the case for the Whitt approximations. Generally for a utilization, $\rho \le 0.5$, Whitt's approximations are not very accurate. I assume that a person designing a system desires maximum efficiency of his system with as little idle time as possible. This is not unreasonable, so I chose a utilization of $\rho \ge 0.9$ for my experiments. Whitt's approximations are limited to the GI/G/m queueing systems with **m** parallel servers and a

3 Gross [11], p. 423.
4 Gross [11], p. 420.

First Come — First Served discipline. The approximations use four basic parameters:

i) the squared coefficient of variation for the arrivals $c_a^2 = \frac{\sigma^2_{\text{arrival}}}{1/\lambda^2}$;

ii) the squared coefficient of variation for the service $c_s^2 = \frac{\sigma^2_{\text{service}}}{1/\mu^2}$;

iii) the utilization (traffic intensity) $\rho = \frac{\lambda}{m\mu}$;

iv) the number of parallel servers m;

Whitt uses the squared coefficients of variation instead of the variance because "they are dimensionless and it is easily interpreted independently of the mean"[5]

The basic approximation used as the true expected wait time in the queue is:

$$E\left[W_{Q_{GI/G/m}}\left(\rho, c_a^2, c_s^2, m\right)\right] = \phi\left(\rho, c_a^2, c_s^2, m\right)\left(\frac{c_a^2 + c_s^2}{2}\right)E\left[W_{Q_{M/M/m}}\right]$$

where
$$\phi\left(\rho, c_a^2, c_s^2, m\right) = \begin{cases} \left(\frac{4\left(c_a^2 - c_s^2\right)}{4c_a^2 - 3c_s^2}\right)\phi_1(m,\rho) + \left(\frac{c_s^2}{4c_a^2 - 3c_s^2}\right)\Psi\left(\frac{c_a^2 + c_s^2}{2}, m, \rho\right) & c_a^2 \geq c_s^2 \\ \left(\frac{\left(c_s^2 - c_a^2\right)}{2c_a^2 - 2c_s^2}\right)\phi_3(m,\rho) + \left(\frac{c_s^2 + 3c_a^2}{4c_a^2 - 3c_s^2}\right)\Psi\left(\frac{c_a^2 + c_s^2}{2}, m, \rho\right) & c_a^2 \leq c_s^2 \end{cases}$$

$$\Psi\left(\frac{c_a^2 + c_s^2}{2}, m, \rho\right) = \begin{cases} 1 & \frac{c_a^2 + c_s^2}{2} \geq 1 \\ \phi_4(m, \rho)^{2\left(1 - \frac{c_a^2 + c_s^2}{2}\right)} & 0 \leq \frac{c_a^2 + c_s^2}{2} \leq 1 \end{cases}$$

$$\phi_1(m,\rho) = 1 + \gamma(m,\rho) \quad \text{where } \gamma(m,\rho) = \min\left\{0.24, \frac{(1-\rho)(m-1)\left((4+5m)^{0.5} - 2\right)}{16m\rho}\right\}$$

$$\phi_2(m,\rho) = 1 - 4\gamma(m,\rho)$$

$$\phi_3(m,\rho) = \phi_2(m,\rho)\exp\left(\frac{-2(1-\rho)}{3\rho}\right)$$

$$\phi_4(m,\rho) = \min\left\{1, \frac{\phi_1(m,\rho) + \phi_3(m,\rho)}{2}\right\}$$

This approximate value is enough to use in the truncation heuristic, but it is not enough to use in setting the initial conditions. We could apply Little's formula to find an approximate expected number in the queue and round the value to an integer to set the initial conditions. As we discussed before, however, this would only be a deterministic initial

[5] Whitt [40], p. 115.

condition that would ignore other possible states.

Whitt [40] provides approximations for the steady state queue length and number in the system distributions. Like Kelton [17], Whitt uses the geometric distribution as the building block for his queue length distribution. He then checks four different possible cases of the squared coefficient of variation of the conditional queue length (queue length given there is a queue), c_c^2. These four cases result in unique algorithms for generating a queue length.

The first case is when $c_c^2 > 1 - E[Q_L \mid Q_L > 0]^{-1} + 0.02$. The queue length distribution is a mixture of two geometric distributions on positive integers with p1, p2, and γ that match the conditional expected queue length and the squared coefficient of variation of the conditional queue length having balanced means. The approximation equations for case one are:

$$P(C = k) = \gamma p_1 (1 - p_1)^{k-1} + (1-\gamma) p_2 (1 - p_2)^{k-1} \qquad k \geq 1$$

$$P(C > k) = \gamma (1 - p_1)^k + (1-\gamma)(1 - p_2)^k$$

$$\text{where } p_1 = m_1^{-1} > p_2 = m_2^{-1}, \ \frac{E[Q_L \mid Q_L > 0]}{2} = \gamma m_1 = (1-\gamma) m_2 \ \text{ and}$$

$$\gamma = \left[1 + \left(1 - 2\left[c^2 + 1 + \frac{1}{E[Q_L \mid Q_L > 0]} \right]^{-1} \right)^{0.5} \right] \Big/ 2.0$$

The second case is when $\left| c_c^2 - 1 + E[\text{Number in Queue} \mid \text{A Queue Exists}]^{-1} \right| \leq 0.02$. The queue length is simply a geometric distribution with $p = \dfrac{1}{E[Q_L \mid Q_L > 0]}$. The approximation CDF equation is: $P(C = k) = p(1-p)^{k-1} \quad k \geq 1$

The third case is applicable when

$$\frac{\left(E[Q_L \mid Q_L > 0]^2 - 1 \right)}{2E[Q_L \mid Q_L > 0]^2} < c_c^2 \leq 1 - \frac{1}{E[Q_L \mid Q_L > 0]^2 - 1} - 0.02.$$

The queue length distribution is a convolution of two geometric distributions. The first distribution is across nonnegative integers with mean $m_1 = (1 - p_1)/p_1 \geq 0$; the second across positive integers with mean $m_2 = 1/p_2 \geq 1$. The approximation equations for case three are:

$$P(C=k)=\sum_{j=0}^{k-1} p_1(1-p_1)^j p_2(1-p_2)^{k-j-1} \qquad k\geq 1$$

$$P(C>k)=1-\sum_{j=0}^{k} P(C=j)$$

$$m_1 = \frac{\left((x-1)-\left[(x-1)^2-2x^2\left(1-c^2-x^{-1}\right)\right]^{0.5}\right)}{2} \qquad p_1=\frac{1}{m_1+1}$$

$$m_2 = \frac{\left((x+1)+\left[(x-1)^2-2x^2\left(1-c^2-x^{-1}\right)\right]^{0.5}\right)}{2} \qquad p_2=\frac{1}{m_2}$$

The final case is when $c_c^2 > \frac{\left(\left(E[Q_L \mid Q_L>0]\right)^2-1\right)}{2\left(E[Q_L \mid Q_L>0]\right)^2}$. The equations for case four are essentially the same as for case three with the exception:

$$m_1 = \frac{\left(E[Q_L \mid Q_L>0]-1\right)}{2} \qquad p_1=\frac{1}{m_1+1}$$

$$m_2 = \frac{\left(E[Q_L \mid Q_L>0]+1\right)}{2} \qquad p_2=\frac{1}{m_2}.$$

The algorithms to generate a queue length from these four cases are at Appendix A.

The approximation for the distribution for the number in the system is:

$$P(N=k) = \begin{cases} P(Q=k-m) & k\geq m+1 \\ p(k) & 0\leq k\leq m \end{cases}.$$

The p(k) is a truncated Poisson distribution with intensity α. Whitt considers the truncated Poisson distribution because the number of customers in the system for the GI/G/∞ Queueing system is asymptotically normally distributed. The Poisson distribution is a reasonable discrete analog. Whitt notes that this approximation should work extremely well for greater number of servers.

Whitt's derived approximations allow me to test the hypothesis that using the approximation I can control the initial condition bias and get a more precise and more accurate estimate even with a maximum of four servers.

CHAPTER 3. METHODOLOGY

This chapter provides insight to my techniques. I first develop comprehension of some critical topics, before I illustrate my heuristic methodologies. These topics are:

i) Replication/deletion and Batch Means methods for producing independent estimates.

ii) Variance reduction through Common Random Number and Synchronization.

iii) Estimators of a desired statistic.

I then discuss setting the initial conditions in a simulation. Next, I provide the framework the truncation heuristics and show some interesting preliminary results from my research. The last section of this chapter is an description of the experiments I used to evaluate the performance of the heuristics.

3.1. Batch Means versus Replication Deletion

We know that a simulation output sequence is sequentially correlated. To effectively analyze any estimator, we want to perform a statistical analysis upon our estimator to create a confidence of our methodology. Accordingly, we want to try and get independent estimates from a simulation model; from which we can perform statistical analysis. There are two methods to create independent estimates from simulation output sequences: Batch Means and Replication/deletion.

Batch Means is based upon a single long simulation run. The advantage of using batch means is the sequence has to travel through a transient period only once. "Suppose that we make a simulation run of length m and divide the resulting observations $Y_1, Y_2, \ldots, Y_m$, into n batches of length k. (Assume that $m = nk$) Thus, batch 1 consists of observations $Y_1, Y_2, \ldots, Y_k$, batch 2 consists of observations $Y_{k+1}, Y_{k+2}, \ldots, Y_{2k}$, etc. Let $\overline{Y}_j(k)$ (where $j = 1, 2, \ldots, n$) be the sample (or batch) mean of the k observations in the jth batch and let

$$\overline{\overline{Y}}(n,k) = \frac{\sum_{j=1}^{n} \overline{Y}_j(k)}{n} = \frac{\sum_{i=1}^{m} Y_i(k)}{m}$$

be the grand sample mean."[6] The function $\overline{\overline{Y}}(n,k)$ becomes the point estimator for θ. The idea is that the batch means will be far enough away from each other that the observations will be independent. A major concern is whether or not a single run ***is*** long enough to fully go through the transient period. Often we are constrained to a finite computer budget. If the transient period is longer than the single long run, there exists the possibility that the resulting $\overline{\overline{Y}}(n,k)$ is extremely biased and the observations used to determine the estimate are correlated.

Replication/deletion is easier to use and implement that Batch Means. Instead of one long run, replication/deletion takes the estimates of several shorter runs to develop a point estimate for θ. Consider $X_1, X_2, \ldots\ldots, X_n$, to be the end run sample means of our desired performance parameter. These occur from n simulation runs. Each run has a unique sample path. The result is n independent observations of what we are trying to estimate. Our estimator for θ simply becomes $\overline{X}(n) = \sum_{i=1}^{n} X_i / n$. The problem with replication/deletion is each replication must go through a transient phase and the run lengths are shorter.

There is considerable debate whether it is better to use batch means for a point estimate of replication deletion. I do not intend to debate this issue. I apply replication deletion and batch means analysis to show the applicability to both methods.

3.2. Variance Reduction and Synchronization

To reduce the variability in our analysis, we attempt to **synchronize** the simulation models by inducing positive correlation between the systems we are comparing. That is if we are to compare one model to another, we must ensure it is under "like" conditions. In this research, I focus on the estimation of the expected wait time in the queue. I compare the stochastically set initial condition system to an empty and idle system. I do so by implementing common random numbers (CRN). "To implement CRN properly, we must

---

6 Law & Kelton [20], p.554.

match up, or *synchronize*, the random numbers across different system configurations on a particular replications. *Ideally*, a specific random number used for a specific purpose in one configuration is used for *exactly the same* purpose in all other configurations."[7]

When my simulation model initializes the random variables at simulation time 0, my initial condition block stochastically generates a random initial number in the system based upon Whitt's approximate distributions discussed earlier. These "customers" each have a unique service time. Additionally, a queue can exist at start up. To compare the estimate of the stochastically set system to an empty and idle system, I need to perform two tasks to ensure synchronization. First, I need to record only the wait time observations of those "customers" generated after simulation startup. Secondly, I must ensure that the service time the first generated "customer" in the empty and idle system has the same service time that the first generated "customer" received after all the stochastically generated customers have been served. Synchronization is critical in analyzing experimental results. Consider a stochastically set initial number in the system is 5 for a single server queue. If you do not synchronize the service times for the first randomly generated customer, it will receive the first stochastically set customer's service time. By synchronizing the service times between the systems the estimate comparison of the wait in the queue now has no variability that would have resulted in an unsynchronized system. To accomplish synchronization, I recorded the seed value of the service time when the stochastically set number in the system is finished being served. I then ensured the empty and idle system had the synchronized service time for the generated customers by setting the initial server seed value to the recorded value. In doing so I eliminated any variability that would occur by using different service times in my comparison of stochastically set initial conditions models and empty and idle initial condition models.

3.3. Estimators of a Performance Parameter

An estimator is a mathematical function which provides a point estimate value of a

[7] Law & Kelton [20], p.617.

true value. An estimate is the value an estimator produces, given an input. An estimator of the expected value of a sample data set is typically the average of that data set. By the Law of Large Numbers, as the number of observations approaches infinity, the sample mean will asymptotically approach the true expected value; $\hat{\theta} \underset{n \to \infty}{\rightarrow} \theta$. A problem arises when the observations used in the estimator are sequentially correlated. This is typically the problem in a simulation output sequence. In queueing simulations, a large observation is typically followed by another large observation. This makes it difficult to estimate the variance of the estimator.

I developed several estimators of the expected wait in the queue using Whitt's approximation for the steady state expected wait in the queue. The estimators are in essence an average. What makes them different from a cumulative run mean is the heuristic seeks out "better" sub-sequence of data to provide my estimate. These estimator, thus, are functions of a random truncation point in the output sequence. **Table 3.1**, on page 38, lists the estimator for each heuristic.

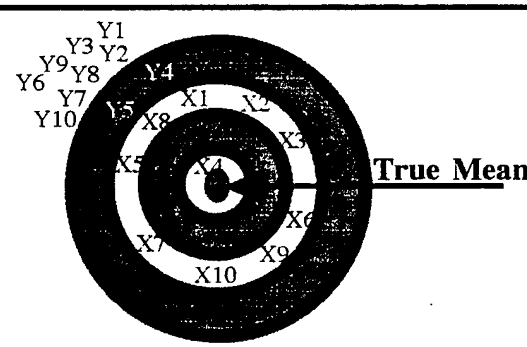
3.4. Initial Conditions

Setting the initial conditions in a queueing model may allow the simulation to have a shorter transience phase. This is theoretically sound since it attempts to establish an initial condition representative of the asymptotic state. Since we cannot achieve an asymptotic state in a finite simulation run, simply changing the starting state from empty and idle to another deterministic condition does nothing more than what the empty and idle condition does at the outset of the simulation. It ignores other possible starting states. Some argue that empty and idle is an asymptotic state. While I concede there is some probability of its occurrence, assuming that it occurred at the outset with probability 1.0 may not be representative of the asymptotic distribution across states. Additionally, assuming a non-zero deterministic initial condition is not representative either.

Stochastically setting the initial conditions allows the model to draw from an approximated distribution to set the initial conditions. The theory is, if the approximation is close to the true distribution, then across simulation runs, initial conditions are chosen according to a distribution which is close to the actual steady state distribution. Kelton [17] illustrates how we can use geometric and uniform distributions to stochastically set the initial conditions with good results. Whitt [40] derives approximations for the steady state number in the queue and in the system distributions. I use Whitt's approximations in my experiments.

3.5. Overview and the MSEAT Truncation Heuristic

As I discussed earlier, if we knew the true expected value of a performance parameter before a simulation, then there would be little need to simulate. We rarely have this knowledge of the system. We must know distributions in order to simulate. Often, we make an assumption as to which distribution to use if we do not have historical data. We want to use as much apriori information as possible to ensure the output estimate is both accurate and precise. I considered Fishman's [6] penalty of truncation using the MSE equation. I believe we can use the minimum estimated MSE as a truncation point where the minimum penalty occurs and results in a precise and accurate estimate.



$\overline{X}$ = More accurate estimate but with less precision than $\overline{Y}$. This equates to greater variance but less bias.

$\overline{Y}$ = More precise estimate but with less accuracy than $\overline{X}$. This equates to less variance but greater bias.

Figure 3.1. Target Estimate of Simulation Output

My methodology differs from other truncation heuristics, in that I assume no knowledge of the system based upon pilot runs. Indeed, I eliminate pilot runs completely.

The first and second moment information from the arrival and service distributions partially specifies the underlying distribution of the estimated performance parameter. Whitt [40] shows that derived approximations for the long-run distribution of the number in a system and in a queue are quite accurate; with the worst absolute error being about 20%. An approximation of even 20% absolute error to the true value can assist us in obtaining a "better" estimate when compared to simply accepting a simulation run mean. I define the term "better" estimate to be an estimate that is both precise and accurate. **Figure 3.1.** illustrates how precision and accuracy may vary in sample data.

The estimated MSE,

$$\begin{aligned} \text{MSE} &= \text{E}\left[\left(\hat{\theta}-\theta\right)^2\right] \\ &= \text{Bias}^2\left[\hat{\theta}\right]+\text{Var}\left[\hat{\theta}\right] \end{aligned}$$

where $\hat{\theta}$ is an estimator of θ, is one statistical computation that considers both the accuracy (bias) and the precision (variance) of the estimator. An estimate of the MSE of the estimator is:

$$\hat{\text{MSE}}\left[\hat{\theta}\right] = \hat{\text{Bias}}^2\left[\hat{\theta}\right]+\hat{\text{Var}}\left[\hat{\theta}\right].$$

$$\hat{\text{Bias}} = \overline{\text{X}}-\theta \qquad \hat{\text{Var}} = \hat{\sigma}^2 = \frac{\sum_{i=1}^{n}\left(X_i-\theta\right)^2}{(n)^2}$$

where θ = the true mean and X_i is an observation of the estimate independent of other observations You can see that not knowing the true mean precludes you from using the MSE as a truncation heuristic. The best you can do is derive an estimate of the variance of the estimator using the sample variance equation: $S^2 = \frac{\sum_{i=1}^{n}\left(X_i-\overline{X}\right)^2}{(n-1)}$. Assuming independent data, then $\hat{\sigma}^2 = \frac{S^2}{n} = \frac{\sum_{i=1}^{n}\left(X_i-\overline{X}\right)^2}{(n-1)n}$ is an unbiased estimator for $\text{Var}\left[\overline{\text{X}}\right]$. "Bias and variance measure two separate characteristics of an estimator. Bias measures systematic deviation from the true mean, whereas variance measures variation around the bias plus the mean."[8] The goal is to make best use of the simulation output sequence in estimating the true expected value of the unknown distribution.

I consider using the point approximation for the expected steady state wait time in the queue Whitt [40] develops as the true analytical mean to compute an estimate of the MSE. Using the approximate mean as the true mean allows me to compute an estimate of the sample variance without the use of a sample mean. $\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}\left(X_i - \theta_{approx}\right)^2}{n}$

This gives me as an estimate for the variance of the estimator $\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}\left(X_i - \theta_{approx}\right)^2}{n^2}$.

At the minimum estimated MSE point,

$$d_{trunc} = \min\left[\hat{MSE}\right] = \underset{n>d\geq 0}{\arg\min}\left[\left(\overline{X}_{n,d} - \theta_{approx}\right)^2 + \frac{\sum_{i=d+1}^{n}\left(X_i - \theta_{approx}\right)^2}{(n)^2}\right],$$

I discard the previous data (truncate) and calculate an average of the remaining observations as the point estimate. My final equation for the point estimator is $\hat{\theta} = \frac{\sum_{i=d+1}^{n} X_i}{n-d}$, where "d" is the point of minimum estimated MSE and X_i is the "*i*th" observation of the parameter I am estimating. I call this truncation heuristic Mean Square Error Approximation Truncation (MSEAT).

Using the minimum estimated MSE as a truncation point allows me to do the best I can from a finite run length output sequence in obtaining an accurate and precise estimate.

Figure 3.2. illustrates how the minimum estimated MSE truncation works. The minimum estimated MSE point yields the truncation point. You can see that each sample path has a unique behavior. The top left sample path has a truncation point at the 1999th observation. The remaining data provides the point estimate. Likewise, the top right and bottom sample paths have truncation points of 9706 and 913, respectively. Concerns I have for my methodology are the strength and sensitivity of using the approximation as the true mean. If the approximation has significant error to the true expected value, then the point estimate of the performance parameter from truncation may be worse than a run

---

8 Fishman [6], p.787.

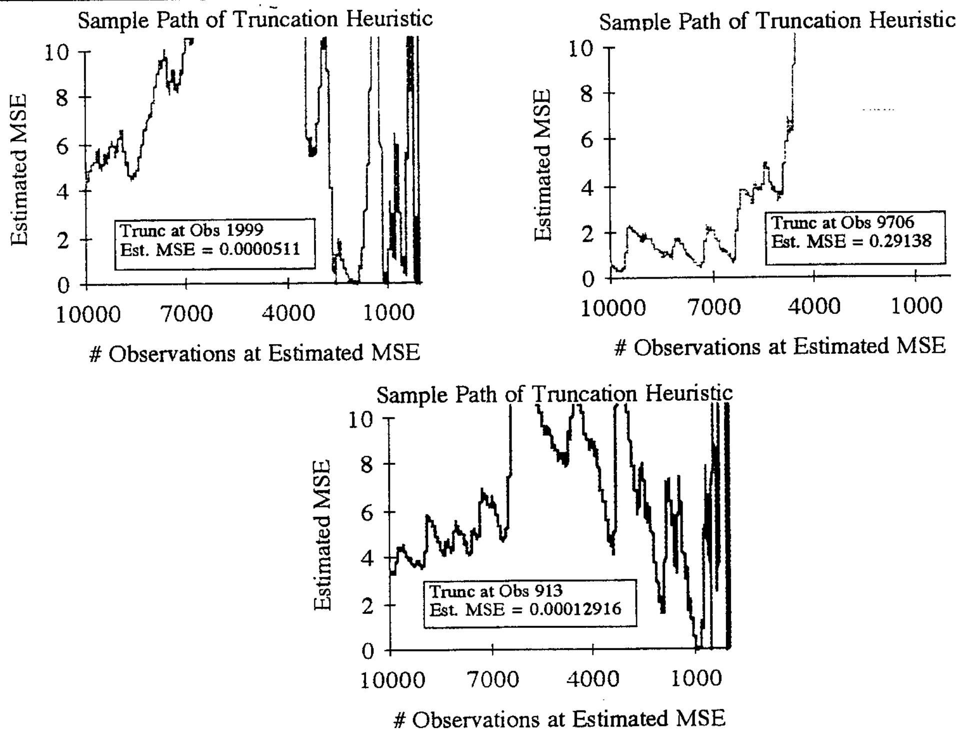


Figure 3.2. Illustrations of Estimated MSE Truncation Heuristic

mean. To test this concern, I ran 41 experiments of 30 replications with 10,000 customers exiting the system of an M/M/1 queue with common random numbers. I used the true analytical expected wait time in the queue as the initial approximation and then incremented the value by ±0.01 absolute error to give an absolute error from 0 to 20% for both positive and negative error. Note that Whitt's [40] approximations reduce to the exact true M/M/1 values.

Figure 3.3 depicts the findings across the 41 experiments. The significant result was that even with an approximation absolute error of 20% the truncation estimate was more precise and accurate than the run mean with no truncation. The fact that the average across 30 sample paths could not produce an estimate within the 95% confidence lower bound of the approximation heuristic is even more significant. This is attributable to the negative initialization bias M/M/1 queues experience when starting with empty and idle initial conditions and the extremely long transient period associated with the M/M/c models.

However, this test also shows the significant effectiveness of using apriori information to our advantage in an approximation assisted point estimate.

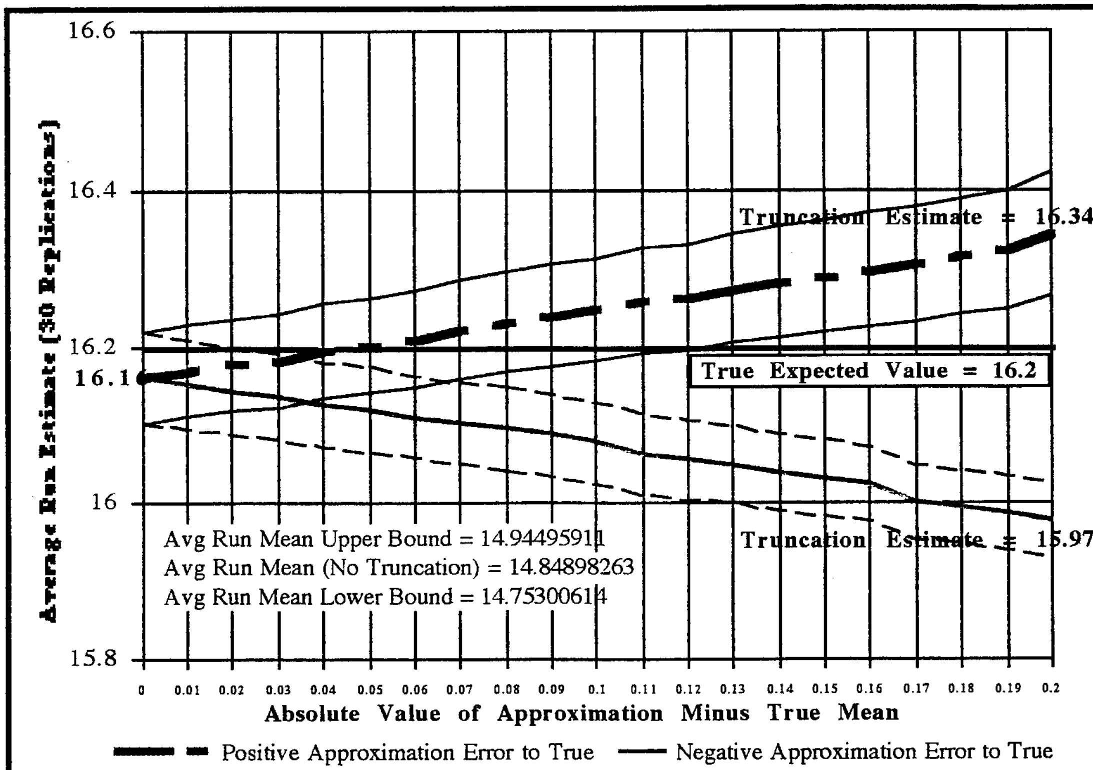


Figure 3.3 Approximation Sensitivity

3.5.1. MSEAET Heuristic

I noted that the strength of the approximation in the bias and variance equations forced the estimate towards the approximation value. I considered "softening" the strength of the heuristic to let the sample data have a more determinant impact on the estimate. I felt that maintaining the bias equation and using a different equation for the variance would lessen the strength of the heuristic, but still allow for better precision and accuracy.

Since the power of the heuristic appeared to force the estimate toward the approximate value, I considered using this power to my advantage. If I knew whether the approximation had a negative or positive error to the true expected value, I could weight the approximation to negate the direction of the error. I do not believe this is information the user could provide. For my initial $E_2/E_2/4$ experiments with $\rho = 0.90$, I input the absolute error of the approximation to generate two new approximations that I then used in a modification of the MSEAT heuristic The approximation equations are:

$$\theta_{approx_1} = \theta_{approx}; \quad \theta_{approx_2} = \theta_{approx} + |ApproxError|; \quad \theta_{approx_3} = \theta_{approx} - |ApproxError|;$$

I then found three separate truncation points as I did in the MSEAT heuristic and calculated an estimate from each truncation point. I then took the average of these estimates as my point estimate. I call this heuristic the Means Square Error with Approximation Error Truncation (MSEAET) heuristic. Intuitively, we can see for my initial $E_2/E_2/4$ experiment that either θ_{approx_2} or θ_{approx_3} is the true expected wait in the queue depending on the direction of the approximation error. I am concerned that the resulting estimates will be biased in the direction of the approximation error since this heuristics weights the direction of the error. Nevertheless, it has sound justification for investigation. Since the size and direction of the error are information the simulator would not know, I use 0.20 as the approximation error for the remainder of my experiments. I use this value since I have already shown that a 20% absolute error can produce a "good" estimate and Whitt's approximations are, on average, more accurate than this.

The final equations for the MSEAET heuristic estimator are:

$$\hat{\theta}_1 = \frac{\sum_{i=d_1+1}^{n} X_i}{(n-d_1)} \quad \hat{\theta}_2 = \frac{\sum_{i=d_2+1}^{n} X_i}{(n-d_2)} \quad \hat{\theta}_3 = \frac{\sum_{i=d_3+1}^{n} X_i}{(n-d_3)} \quad \hat{\theta} = \frac{\hat{\theta}_1 + \hat{\theta}_2 + \hat{\theta}_3}{3.0},$$

where d_1, d_2, and d_3 are shown in **Table 3.1.**

3.5.2. MSEASVT Heuristic

White[36] describes a heuristic of minimizing the sample half-width in simulation output data as a method of truncation. While this indeed maximized the precision of the estimate, I found that using it to determine an estimate of the expected wait time in the queue occasionally resulted in extremely biased estimates when the sample paths had small and similar observations at the tail end of a simulation. I considered that using the White's sample data precision with the original approximation bias equation would lessen the pull of the truncation heuristic to the approximation. I combined White's half-width equation with the approximation bias equation in estimating the MSE to determine a suitable truncation point: $\left(\widehat{\text{MSE}} = \left(\overline{X}_{n,d} - \theta_{approx}\right)^2 + \frac{\sum_{i=1}^{n} X_i^2 - (n-d)\overline{X}_{n,d}^2}{(n-d)^2(n-d)} \right)$, where "d" is the truncation point.

3.5.3. MSESET Heuristic

One final heuristic I considered is applicable to any simulation. The idea is simply to perform a back-end analysis of the sample data by calculating a run sample mean and a standard error using a desired α level from only the last half of the sample data. I used the sample means from the second half of the output sequence as the true mean and perturbed it by the standard error which gave me three values to use in my estimation of the MSE. Similar to the MSEAET, I then analyze the output sequence for the point with the minimum estimated MSE and find three truncation points. I then calculate three estimates of the wait time in the queue and use the average of the three as my point estimate. For my experiments I used an α of 0.10 to calculate a 90% confidence standard error.

Using the last half of the sample data to compute a "true" expected value assumes that the run length was long enough that the transient period occurred in the first half of the sample data and the second half of the output sequence is more representative of the steady state distribution. While this may not be the case, it does lend itself for interesting debate and a desire on my part to see what happens. **Table 3.1.** summarizes the heuristics.

3.6. Experiments Overview

To test the heuristics, I set up a series of non-standard queueing models with a traffic intensity $\rho \geq 0.9$. I used models other than the M/M/c model since the Whitt approximations reduce to the M/M/c case. To check my results I used Queueing Tables and Graphs, by Hillier and Yu and Tables for Multi-Server Queues, by Seelen, Tijms and Van Hoorn. I analyzed the models stochastically setting the initial conditions and then, with recorded seeds, empty and idle. I performed this analysis for batch means and replication

Heuristic	Truncation Point (d) Equations	Expected Value Estimator
MSEAT	$d_{trunc} = \underset{0 \le d < n}{\arg\min}\ \hat{\text{Bias}}^2\left[\hat{\theta}\right] + \hat{\text{Var}}\left[\hat{\theta}\right]$ $\hat{\text{Bias}}\left[\hat{\theta}\right] = \overline{X}_{d,n} - \theta_{approx}$ $\hat{\text{Var}}\left[\hat{\theta}\right] = \hat{\sigma}^2 = \dfrac{\sum_{i=d+1}^{n}\left(X_i - \theta_{approx}\right)^2}{(n-d)^2}$ $\hat{\text{MSE}}\left[\hat{\theta}\right] = \left(\overline{X}_{d,n} - \theta_{approx}\right)^2 + \hat{\sigma}^2$	$\hat{\theta} = \dfrac{\sum_{i=d+1}^{n} X_i}{(n-d)}$
MSEAET	$d_{trunc} = \underset{0 \le d < n}{\arg\min}\ \hat{\text{Bias}}^2\left[\hat{\theta}\right] + \hat{\text{Var}}\left[\hat{\theta}\right]$ $\hat{\text{Bias}}\left[\hat{\theta}\right] = \overline{X}_{d,n} - \theta_{approx}$ $\hat{\text{Var}}\left[\hat{\theta}\right] = \hat{\sigma}_j^2 = \dfrac{\sum_{i=d+1}^{n}\left(X_i - \theta_{approx}\right)^2}{\left(n-d_j\right)^2}$ (j = 1, 2, 3) $\hat{\text{MSE}}_1\left[\hat{\theta}\right] = \left(\overline{X}_{d_1,n} - \theta_{approx}\right)^2 + \hat{\sigma}_1^2$ $\hat{\text{MSE}}_2\left[\hat{\theta}\right] = \left(\overline{X}_{d_2,n} - \left(\theta_{approx} + \text{ApproxError}\right)\right)^2 + \hat{\sigma}_2^2$ $\hat{\text{MSE}}_3\left[\hat{\theta}\right] = \left(\overline{X}_{d_3,n} - \left(\theta_{approx} - \text{ApproxError}\right)\right)^2 + \hat{\sigma}_3^2$	$\hat{\theta}_1 = \dfrac{\sum_{i=d_1+1}^{n} X_i}{\left(n-d_1\right)}$ $\hat{\theta}_2 = \dfrac{\sum_{i=d_2+1}^{n} X_i}{\left(n-d_2\right)}$ $\hat{\theta}_3 = \dfrac{\sum_{i=d_3+1}^{n} X_i}{\left(n-d_3\right)}$ $\hat{\theta} = \dfrac{\hat{\theta}_1 + \hat{\theta}_2 + \hat{\theta}_3}{3.0}$
MSEASVT	$d_{trunc} = \underset{0 \le d < n}{\arg\min}\ \hat{\text{Bias}}^2\left[\hat{\theta}\right] + \hat{\text{Var}}\left[\hat{\theta}\right]$ $\hat{\text{Bias}}\left[\hat{\theta}\right] = \overline{X}_{d,n} - \theta_{approx}$ $\hat{\text{Var}}\left[\hat{\theta}\right] = \hat{\sigma}^2 = \dfrac{\sum_{i=d+1}^{n}\left(X_i - \overline{X}\right)^2}{(n-d)(n-d-1)}$ $\hat{\text{MSE}}\left[\hat{\theta}\right] = \left(\overline{X}_{d+1,n} - \theta_{approx}\right)^2 + \hat{\sigma}^2$	$\hat{\theta} = \dfrac{\sum_{i=d+1}^{n} X_i}{(n-d)}$
MSESET	$d_{trunc\,j} = \underset{0 \le d < n}{\arg\min}\ \hat{\text{Bias}}^2\left[\hat{\theta}\right] + \hat{\text{Var}}\left[\hat{\theta}\right]$ $\text{Bias}\left[\hat{\theta}\right] = \overline{X}_{d,n} - \overline{X}_{n/2,n}$ $\text{Var}\left[\hat{\theta}\right] = \hat{\sigma}_j^2 = \dfrac{\sum_{i=d+1}^{n}\left(X_i - \overline{X}_{n,d}\right)^2}{(n-d)(n-d-1)}$ (j = 1, 2, 3) $\text{StdError}\left[\hat{\theta}\right] = \sqrt{\dfrac{\hat{\sigma}^2}{n/2}} = \sqrt{\dfrac{\sum_{i=n/2}^{n}\left(X_i - \overline{X}_{n/2,n}\right)^2}{\left(n/2 - 1\right)\left(n/2\right)}}$ $\hat{\text{MSE}}_1\left[\hat{\theta}\right] = \left(\overline{X}_{n,d_1} - \overline{X}_{n/2,n}\right)^2 + \hat{\sigma}_1{}^2$ $\hat{\text{MSE}}_2\left[\hat{\theta}\right] = \left(\overline{X}_{n,d_2} - \left(\overline{X}_{n/2,n} + 1.645(\text{StdError})\right)\right)^2 + \hat{\sigma}_2{}^2$ $\hat{\text{MSE}}_3\left[\hat{\theta}\right] = \left(\overline{X}_{n,d_3} - \left(\overline{X}_{n/2,n} - 1.645(\text{StdError})\right)\right)^2 + \hat{\sigma}_3{}^2$	$\hat{\theta}_1 = \dfrac{\sum_{i=d_1+1}^{n} X_i}{\left(n-d_1\right)}$ $\hat{\theta}_2 = \dfrac{\sum_{i=d_2+1}^{n} X_i}{\left(n-d_2\right)}$ $\hat{\theta}_3 = \dfrac{\sum_{i=d_3+1}^{n} X_i}{\left(n-d_3\right)}$ $\hat{\theta} = \dfrac{\hat{\theta}_1 + \hat{\theta}_2 + \hat{\theta}_3}{3.0}$

MSEAT = **Mean Square Error w/Approximation Truncation**
MSEAET = **Mean Square Error w/Approximation Error Truncation**
MSEASVT = **Mean Square Error w/Approximation and Sample Variance Truncation**
MSESET = **Mean Square Error w/Sample Error Truncation**

Table 3.1. Truncation Heuristic Summary

deletion. **Table 3.2.** is a synopsis of the experiments. My initial batch means experiments showed that the truncation heuristics deleted extreme amounts of data. I wanted to see how the truncation heuristics would perform if I did not allow it to search the entire output sequence. I hypothesized that the first half of the data would be more transient than the second half. Accordingly, I performed the MSEAT and MSEAET truncation heuristics on the first half of the data set only. For the batch means $E_2/E_2/4$ models with $\rho = 0.98$, I performed the MSEASVT analysis two ways. I did the analysis as I had done before allowing the heuristics to search the entire output sequence for the estimated minimum MSE. I also performed this analysis on only the first half of the output sequence. I performed a dual analysis for the MSEASVT since it seemed to, overall, perform better than the other two approximation heuristics. The MSEASVT was less aggressive in truncating data, as well.

Model#	Queueing Model	ρ	Empty & Idle or Init QL	# Reps	# Batches	Obs per Rep	# Exps
1	M/M/1 (Sensitivity)	0.9	Empty & Idle	30	0	10,000	41
2	$E_2/E_2/4$	0.9	Init QL	30	0	10,000	25
3	$E_2/E_2/4$	0.9	Empty & Idle	30	0	10,000	25
4	U/Ln/3	0.9	Init QL	10	0	30,000	25
5	U/Ln/3	0.9	Empty & Idle	10	0	30,000	25
6	$E_2/E_2/4$	0.98	Init QL	10	0	21,000	25
7	$E_2/E_2/4$	0.98	Empty & Idle	10	0	21,000	25
8	M/M/2 M/M/3 Tandem Queue	0.9	Init QL	10	0	30,000	25
9	$E_2/E_2/4$	0.9	Init QL	1	30	210,000	25
10	$E_2/E_2/4$	0.9	Empty & Idle	1	30	210,000	25
11	U/Ln/3	0.9	Init QL	1	10	210,000	25
12	U/Ln/3	0.9	Empty & Idle	1	10	210,000	25
13	$E_2/E_2/4$	0.98	Init QL	1	10	210,000	25
14	$E_2/E_2/4$	0.98	Empty & Idle	1	10	210,000	25

Table 3.2. Table of Experiments

In my initial $E_2/E_2/4$ stochastically set experiments with batch means, I found the smallest batch size was 260 observations per batch. Using MINITAB, I checked the autocorrelation of the 30 batch mean values and found no significant autocorrelation. This does not preclude the possibility that a truncation could occur in the output sequence that would create small batches such that a strong correlation could occur. I chose a smaller number of batches with a greater number of observations per experiment for subsequent models.

Finally, I considered a tandem queueing system model to see how the heuristics performed on a slightly more complex system. Consider a piece of furniture that is being reupholstered. The reupholstering company has five workers. Two can each strip a piece of furniture and the other three can each do the reupholstering. Suppose we want to know the long run expected wait the chair will have before being completed. If we assume exponential interarrival and service rates, we can model the system as a tandem queue of M/M/2 and M/M/3 queues. I chose M/M/m models since they allow an analytical solution for comparison purposes. Analyzing the tandem queues as independent queues, I made use of Little's formula and the standard queueing relationships:

$$L = \lambda W; \quad L_Q = \lambda W_Q$$
$$W = W_Q + \frac{1}{\mu}$$

Where $L = E[\text{Number in System}]$
$W = E[\text{Wait in System}]$
$L_Q = E[\text{Number in Queue}]$
$W_Q = E[\text{Wait in Queue}]$

I used $W_{\text{System}} = \left(W_{Q_1} + \frac{1}{\mu_1} \right) + \left(W_{Q_2} + \frac{1}{\mu_2} \right)$ for the expected wait in the system.

CHAPTER 4. RESULTS OF EXPERIMENTS AND ANALYSIS

This chapter analyzes how the heuristics performed across my experiments. Since my goal was to provided a less biased estimate, I present my results focusing on the frequency of close estimates to the true expected wait in the queue value. I break this chapter into two main sections: Replication/deletion analysis and Batch Means analysis. I provide an analysis for each experiment. In graphs and tables, I illustrate the results for stochastically set and empty and idle initial conditions. At the end of each section I present a histogram of the bias across all experiments. I conclude with a thought provoking discussion on some significant findings I found concerning batch means and the risks associated with using only one sample path to determine a point estimate.

4.1. Replication Deletion Experiments

In the following sections, I analyze the results of the replication/deletion experiments. In each section I have a bar chart which depicts the frequency of estimates that were within $\pm 0.05\theta$ and $\pm 0.02\theta$ ($\pm 0.05\theta$ and $\pm 0.02\theta$ for $\rho = 0.98$ experiments); where θ is the true expected value of the wait in the queue. I chose this analysis since it provides the information a decision maker would most likely want to see. Additionally, below each figure is a table of average statistics for the 25 experiments I ran for both stochastically set and empty and idle initial conditions. The **"Std Dev"** columns are the standard deviation of the 25 values of the half-width and bias, respectively. The **"# Decrease"** column is the number of experiments out of 25 that reduced the half-width of the estimate. The **"Avg d*"** column is the average truncation point for each method. The average percent of the data truncated is in the next column. The final column is a percentage of how often the true expected value of the wait time in the queue is within the 95% confidence interval of the 25 macro-replications. The coverage I achieved was not what I had hoped for. I would have expected that 95% of my experiments would cover the true value. However, the significant reduction in the size of the half-width resulted in estimates that were overall very similar. This directly resulted in an extremely small confidence interval. For this reason, I present the coverage, but do not consider it significant. The tables are an average performance for analysis. To see the results of each individual experiment, I refer you to Appendix B.

4.1.1. $E_2/E_2/4$ Queueing Model (ρ = 0.90)

The $E_2/E_2/4$ Queueing Model with a traffic intensity of 0.90 was my first experiment. From **Figure 4.1.** you can see the significant results the approximation heuristics provide. Regardless if the system initial conditions are set stochastically or empty and idle, the approximation truncation heuristics produced estimates closer to the true expected value significantly more often than did simple run mean estimates. **Figure 4.1.** is significant in that it directly maps the reduction in bias estimates. We can see that the run mean at best produces 64% of its estimates within 2% of the true expected value of the wait time in the queue. This compared to the approximation truncation heuristics' worst percentage of 92% shows that I have reduced the bias considerably for this model. **Table 4.1.** shows the average bias values for each estimate method. The standard deviations that I depict in the tables show throughout that I cannot make any conclusions about the average process. If we were to assume an α of 0.05 and calculate a confidence interval for each method, we would find that the run mean confidence interval is such that it overlaps the upper or lower bounds of the heuristic intervals. This means there are times when the run mean produces estimates with very little bias. Since there is no gap, I can only make general conclusions. We can see, however that there is a decrease in the bias and half-width, in general. We can also see, from **Table 4.1.** that the average half-width of the approximation truncation heuristic methods is extremely small. In other words, for each replication, the heuristics produced very similar estimates. This created 30 independent estimates which were not significantly variable. The half-width associated with these values was small. In fact, out of 25 experiments, the approximation truncation heuristics reduced the half width 25 times. We can also see a drastic reduction in the average bias value.

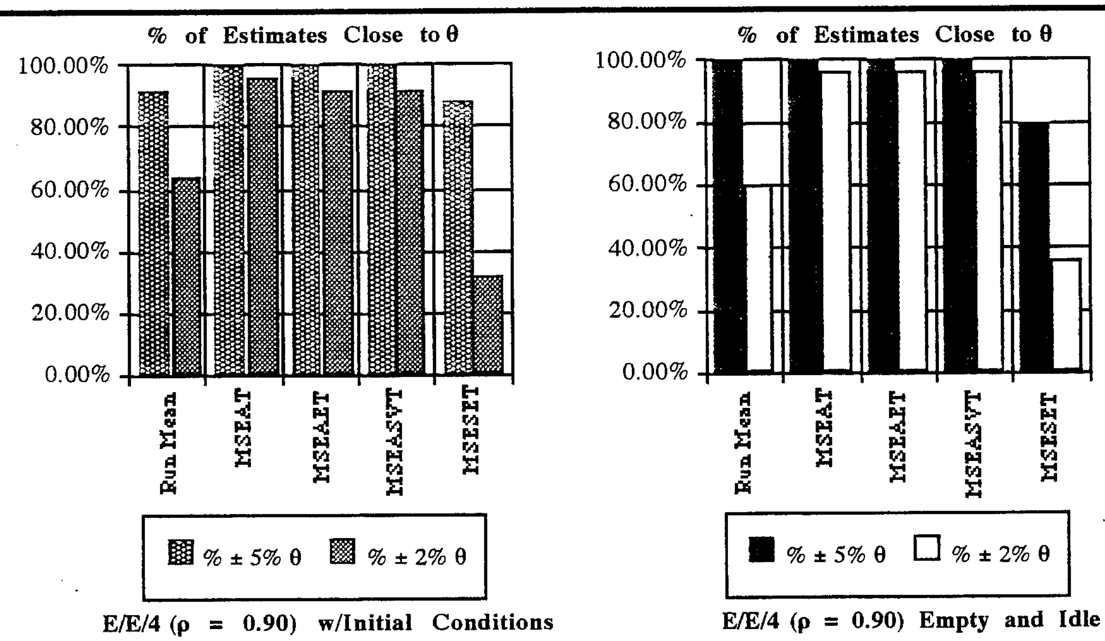


Figure 4.1. Results of $E_2/E_2/4$ (ρ = 0.90) Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	0.8016	0.16	—	0.2671	0.24	92%	64%	0.00	0.00%	96.00%
MSEAT	0.2137	0.12	25	0.1117	0.08	100%	96%	6081.88	60.82%	80.00%
MSEAET	0.2141	0.12	25	0.1183	0.08	100%	92%	5938.28	59.38%	64.00%
MSEASVT	0.2232	0.12	25	0.1268	0.09	100%	92%	4497.01	44.97%	80.00%
MSESET	1.1233	0.22	1	0.4212	0.27	88%	32%	3307.10	33.07%	96.00%

Empty & Idle Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	0.8009	0.15	—	0.2570	0.19	100%	60%	0.00	0.00%	96.00%
MSEAT	0.1832	0.11	25	0.0952	0.08	100%	96%	6151.99	61.52%	60.00%
MSEAET	0.1850	0.11	25	0.0969	0.08	100%	96%	6064.09	60.64%	56.00%
MSEASVT	0.1953	0.10	25	0.1046	0.08	100%	96%	4525.10	45.25%	60.00%
MSESET	1.1336	0.22	0	0.3982	0.30	80%	36%	3329.71	33.30%	92.00%

Table 4.1. Synopsis of $E_2/E_2/4$ (ρ = 0.90) Experiments

On average, the MSESET truncation method did not perform well at all in this model. In fact it increased the bias and half-width. In other words, the MSESET estimate was not accurate nor precise. It is intuitive that the assumption of the second half of the

10,000 observation data set was more representative of the steady state distribution was invalid. I chose to use fewer replications and greater numbers of observations from this point on to see if the theoretical foundation of this heuristic could be supported with better findings.

It is interesting that the average bias for run means of stochastically set models is greater than the empty and idle system. However, as I discussed earlier, the overlapping intervals based on the associated standard error preclude me from making any conclusions. Indeed, if the intervals did not overlap, I would be concerned that the model was incorrect. As it is, I can only conclude that there is statistically no benefit to setting the initial conditions for this model.

An inherent problem with truncation heuristics is they discard data to provide an estimate. The heuristics seek a sub-sequence of the output sequence which is more representative of the true mean. This is interesting in itself, but also lends itself to extreme truncations. Some sample paths truncated as much as 99% of the data to provide the estimate. The reason for this is the transient period associated with a finite run length. If an individual sample path has a long transient period which is also characterized by sever fluctuations, the minimum estimate MSE may not occur until the end of the output sequence. Remember that the heuristics require a sample average. There is no restriction on how large or small the number of observations to produce this sample average is.

The coverage of the estimators was not good. However, we must consider that the half widths are so small, that it's confidence interval associated with the estimator may not include the true expected wait time in the queue. This brings up an important consideration: "Is it better to have estimates closer to the true expected value more frequently, or to include the true expected value in the confidence interval?"

4.1.2. U/Ln/3 Queueing Model

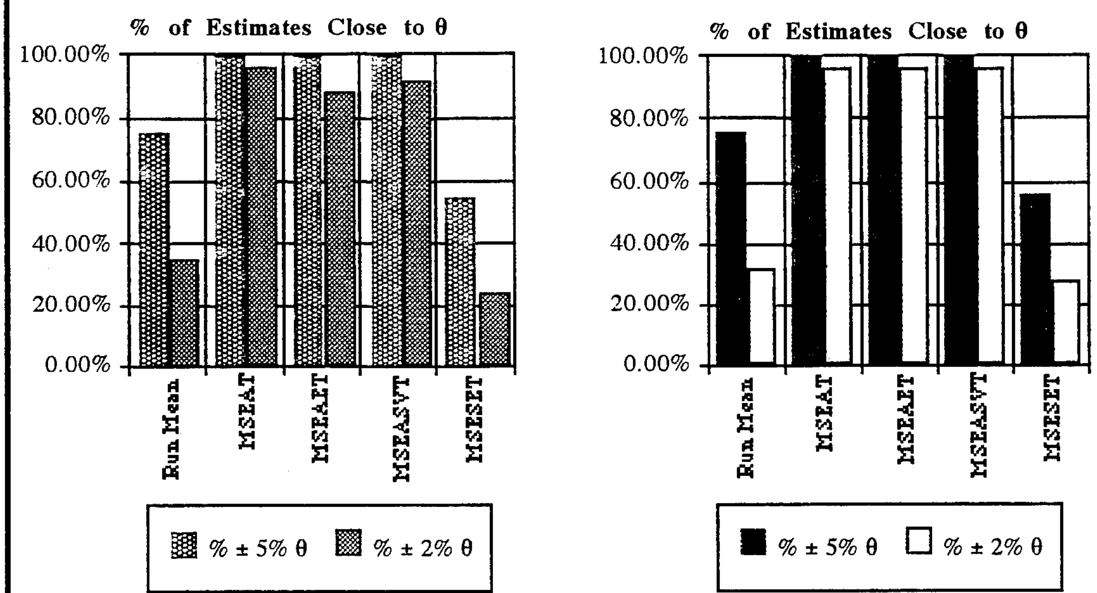
The U/Ln/3 Queueing Model with a traffic intensity of 0.90 was my next experiment. **Figure 4.2.** illustrates that the approximation truncation heuristics once, again, produced exceptional estimates of the true expected wait time in the queue. Again, the run mean estimates, overall, fall short of producing the type of estimates we would desire from our simulation run. The standard error for the bias and the half-width still preclude me from making any conclusion about stochastically setting the initial conditions versus empty and idle initial conditions models.

The MSESET truncation once again did not do well. I consider that possibly increasing the traffic intensity will allow it perform better. At this point, however, I was ready to scrap this methodology. However, it does provide insight into how the sample data actually looks. If the estimate from the MSESET had a negative bias, then we can infer the second half of the output sequence was sequentially correlated low to the true expected value. The converse is true for positive bias.

Similarly to the $E_2/E_2/4$ experiments, 20 out of 25 experiments showed a reduction in the estimate bias for these models for both stochastically set and empty and idle conditions models. In the other 5 experiments, the run mean estimate actually had a closer estimate to the true than the truncation heuristics. However, the truncation heuristic estimates were still very close to the true. The heuristic estimates were worse in these cases since they are using the approximate mean as the true and seek that value out. Since I assume we do not know the true expected value, at this point in time, I cannot think of a way for these cases to accept the run mean over the truncation estimate as a better point estimate. We must consider the overall success of the heuristics, though. They continually produce better point estimates, overall, than do the sample means.

As with the last experiments, I found certain sample paths required a significant truncation of the output sequence. Unless we were able to truly know the length of the transient period, we would not be able to fix this shortcoming. Regardless, the goal is a precise and accurate estimate which the truncation heuristics are producing.

We also see a shortcoming of coverage for these experiments. Again, I do not consider this significant since the accuracy of the estimates forces an exact precision.



U/Ln/3 (ρ = 0.90) w/Initial Conditions **U/Ln/3 (ρ = 0.90) Empty and Idle**

Figure 4.2. Results of U/Ln/3 (ρ = 0.90) Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	0.1194	0.03	—	0.0765	0.04	76%	36%	0.00	0.00%	88.00%
MSEAT	0.0123	0.02	25	0.0240	0.01	100%	96%	18459.43	61.53%	20.00%
MSEAET	0.0408	0.02	25	0.0200	0.02	100%	88%	22288.99	74.30%	84.00%
MSEASVT	0.0189	0.02	25	0.0235	0.01	100%	92%	13281.84	44.27%	32.00%
MSESET	0.1626	0.04	4	0.1049	0.08	56%	24%	9299.12	31.00%	88.00%

Empty & Idle Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	0.1184	0.03	—	0.0745	0.04	76%	32%	0.00	0.00%	92.00%
MSEAT	0.0128	0.02	25	0.0253	0.01	100%	96%	18156.84	60.52%	20.00%
MSEAET	0.0404	0.02	24	0.0200	0.02	100%	96%	21718.26	72.39%	80.00%
MSEASVT	0.0190	0.02	25	0.0240	0.01	100%	96%	12454.99	41.52%	32.00%
MSESET	0.1617	0.04	3	0.1000	0.08	56%	28%	9471.65	31.57%	88.00%

Table 4.2. Synopsis of U/Ln/3 (ρ = 0.90) Experiments

4.1.3. $E_2/E_2/4$ Queueing Model (ρ = 0.98)

In this model, I attempted to create a gap in the standard error interval so I could make a conclusion about setting the initial conditions. As **Table 4.3.** shows, I cannot make any conclusion, once again. Raising the traffic intensity to 0.98 did allow me to show that stochastically setting the initial conditions provided point estimates closer to the true more frequently than empty and idle system. Indeed, as **Figure 4.3.** illustrates, 24 out of 25 experiments, the MSEAT and MSEASVT produced estimates close to the mean when starting the initial conditions stochastically. This is compared to 22 out of 25 experiments that used CRN but were start empty and idle. This is the first time we can make an assumption that as the system becomes more complex, stochastically setting the initial conditions will provide a better estimate than starting a system empty and idle.

The MSESET truncation method continued its poor performance. It continued to provide estimates that were not accurate nor precise. Even increasing the traffic intensity did not help. It is intuitive that the transient period is significantly larger than 30,000. The MSESET heuristic performed a little better for batch means experiments (See Batch Means Analysis). However, it did not do well enough for me to consider it further in my analysis. Accordingly, I only use the MSESET for the U/Ln/3 and $E_2/E_2/4$ ($\rho = 0.90$) Batch Means experiments.

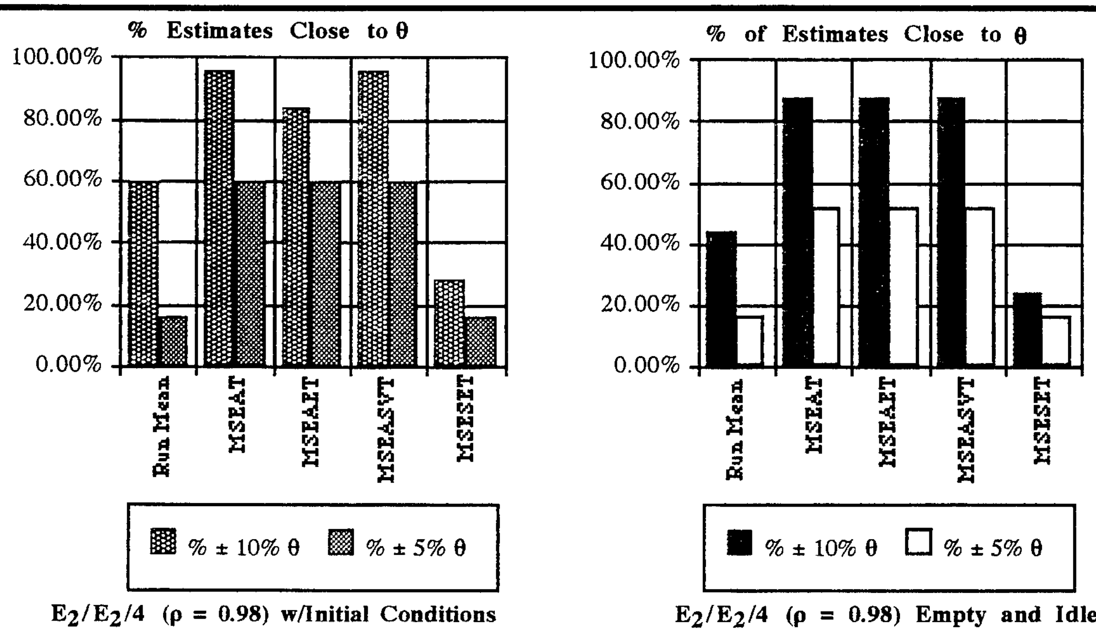


Figure 4.3. Results of $E_2/E_2/4$ (ρ = 0.98) Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	19.9129	6.64	—	11.2715	7.89	60%	16%	0.00	0.00%	76.00%
MSEAT	5.8516	2.58	24	4.0904	3.40	96%	60%	12624.26	60.12%	80.00%
MSEAET	5.9885	2.39	24	4.4655	3.54	84%	60%	12300.38	58.57%	68.00%
MSEASVT	5.8658	2.57	24	4.1233	3.39	96%	60%	12109.57	57.66%	80.00%
MSESET	28.4285	11.73	3	15.4701	9.61	28%	16%	8229.69	39.19%	76.00%

Empty & Idle Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	Avg d*	% Trunc	% Cvg
Run Mean	19.3214	6.37	—	12.0492	7.77	44%	16%	0.00	0.00%	72.00%
MSEAT	6.3261	2.53	24	4.5579	3.61	88%	52%	13242.37	63.06%	68.00%
MSEAET	6.4354	2.40	24	4.8405	3.66	88%	52%	12737.76	60.66%	64.00%
MSEASVT	6.3259	2.50	24	4.5536	3.60	88%	52%	12375.52	58.93%	72.00%
MSESET	28.2734	11.35	3	15.5151	9.49	24%	16%	8259.94	39.33%	76.00%

Table 4.3. Synopsis of $E_2/E_2/4$ (ρ = 0.98) Experiments

4.1.4. Analysis Across Replication Deletion Experiments

I did not achieve the implied coverage of 95% from my replication/deletion experiments using the approximation truncation heuristics. This was a result of extreme precision on the heuristics part. However, as we have seen, I have provided method for replication deletion which reduces the half-width of estimator as well as produce a more accurate estimate of the expected wait time in the queue.

To emphasize how the heuristics performed across experiments, **Figure 4.4** and **Figure 4.5.** depict a histogram of the bias observations for the run mean and MSEAT and MSEASVT truncation heuristics. We can see from **Figure 4.4.** that the majority of the bias observations for the run mean occur in the interval -4.07 to 1.10. Compare this to the MSEAT and MSEASVT central frequency intervals of -1.16 to 0.13 and -1.17 to 0.13, respectively. What this shows us is, in general, the heuristics will produce estimates closer to the true expected value when compared to the run mean. We can further see this by the extreme point on the charts for the run mean. The worst truncation heuristic bias observation is -11.54. Compared to the run mean bias observation of -29.91 or 26.94 extreme points, I have drastically reduced the expected bias of the estimate.

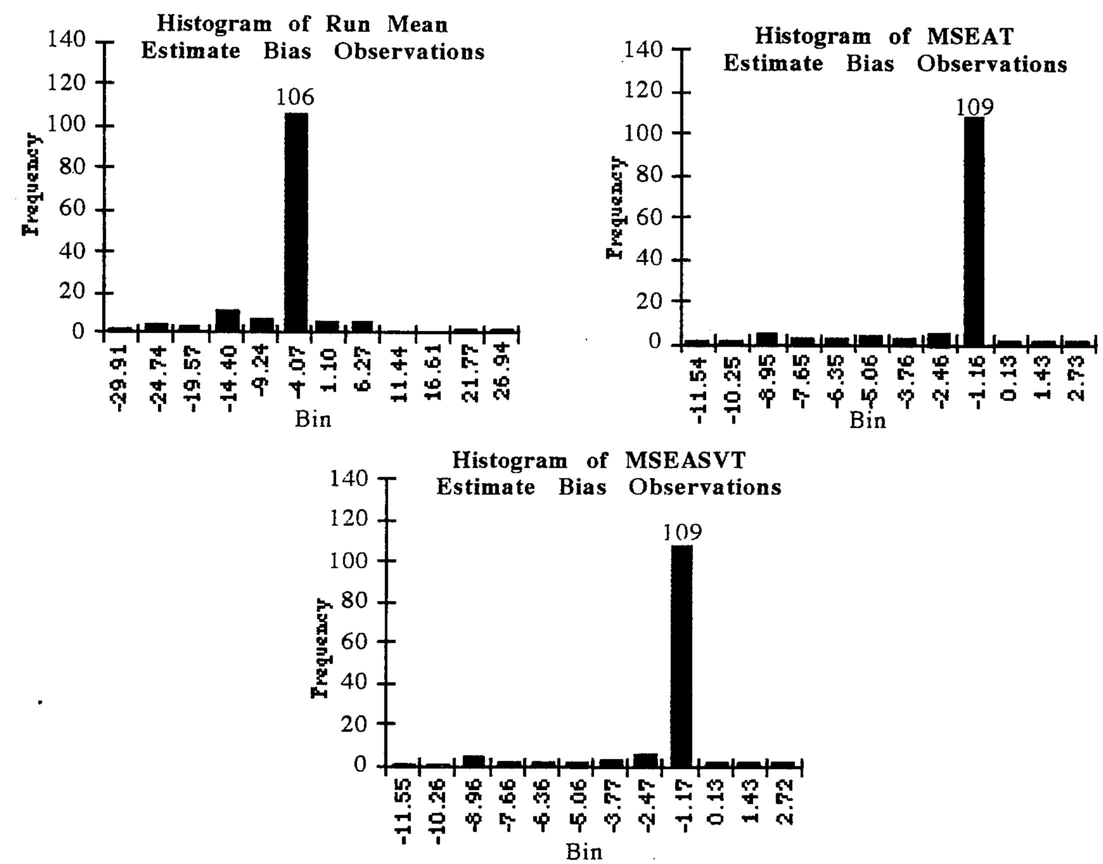


Figure 4.4. Histogram of Replication Deletion Bias Observations

To further expound on this point, I provide a histogram of the central bias observations in **Figure 4.5.** We can see that, once again, the MSEAT and MSEASVT heuristics provide a greater number of estimates closer to the true expected value more frequently than the run mean. Indeed, for the bias interval -0.2 to 0.02, the run mean only produced 58 out of 150 estimates while the MSEAT and MSEASVT produce 96 and 97 out of 150, respectively.

These preliminary findings are encouraging. I now consider the applicability of the truncation heuristics for a batch means analysis. The following section is my analysis for Batch Means. It is followed by my analysis of the Tandem queue model which I performed on a stochastically set initial queue length across replications, only.

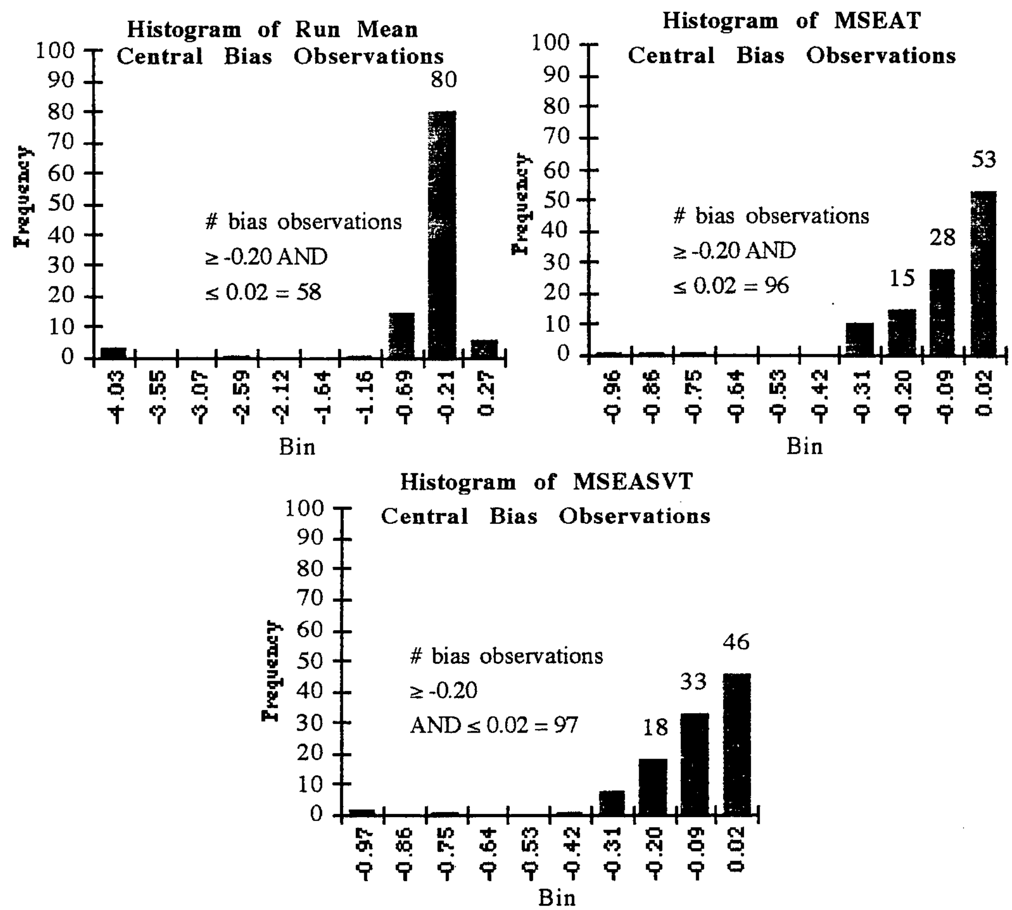


Figure 4.5. Histogram of Replication Deletion Central Bias Observations

4.2. Batch Means Experiments

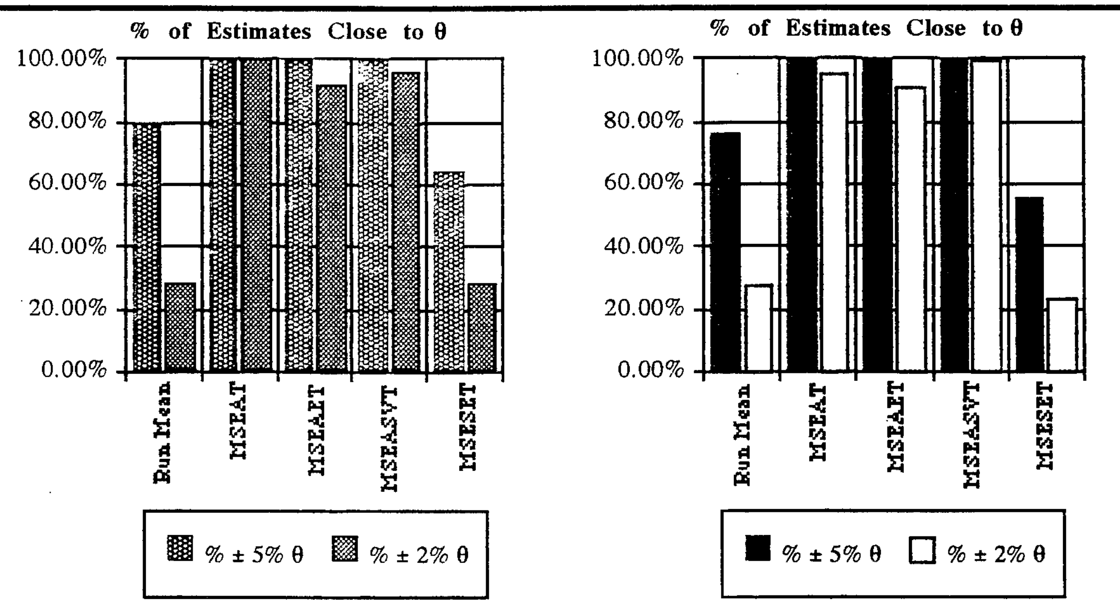
In the following sections, I analyze the results of the batch means experiments. Like the replication deletion analysis, I provide a bar chart which depicts the frequency of estimates that were within ± 0.05θ and ± 0.02θ (± 0.05θ and ± 0.02θ for ρ = 0.98 experiments). Consistent with my replication deletion analysis, below each figure is a table of average statistics for the 25 experiments I ran for both stochastically set and empty and idle initial conditions. The **"d*"** column is ***the*** truncation point the heuristics selected. The average percent of the data truncated is in the next column. Refer to Appendix B for each experiment's results.

4.2.1. $E_2/E_2/4$ Queueing Model (ρ = 0.90)

From **Figure 4.6.** we can see that the heuristics continue to provide excellent point estimates. However, if we examine **Table 4.4.** we can see that we do not find the significant reduction in our half-width. Indeed, the half-width in general increases for all experiments using batch means. This makes sense if we consider my discussion of batch means.

First, the truncation heuristics are searching for a sub-sequence of the simulation output sequence which is more representative of the true expected wait in the queue. Once it finds that sub-sequence, the heuristic batches the remaining data in **"n"** batches. In other words, it divides the sub-sequence and calculates **"n"** estimates of the wait time in the queue. This allows me to build a confidence interval across the batch mean observations, but ultimately results in the point estimate equal to the average of the truncated sub-sequence. This is where we see Fishman's [6] penalty in exact form. We decrease the bias, yet increase the variance. Consequently, we would expect the coverage to increase since the half-width is significantly larger than the replication/deletion analysis. As **Table 4.4.** shows, this is what happened. In one sense, I was pleased that I continued to produce accurate estimates. On the other hand, this increase in the half-width implies a decrease in the precision.

Another concern is the significant amount of the output sequence the truncation heuristics delete. A run length of 210,000 observations equated to approximately 830,000 simulation time increments. As I show later, though considerable in length, the run is not enough.. In general, the MSEASVT was less aggressive than the other approximation heuristics in truncating data.



$E_2/E_2/4$ (ρ = 0.90) w/Initial Conditions

$E_2/E_2/4$ (ρ = 0.90) Empty and Idle

Figure 4.6. Results of $E_2/E_2/4$ (ρ = 0.90) Batch Means Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	0.9808	0.14	—	0.4199	0.27	80%	28%	0.00	0.00%	100 %
MSEAT	2.2461	1.25	1	0.0552	0.01	100%	100%	133176	63.42%	100 %
MSEAET	1.5061	0.49	6	0.1129	0.14	100%	92%	144961	69.03%	100 %
MSEASVT	1.4075	0.64	7	0.0893	0.13	100%	96%	88814	42.29%	100 %
MSESET	0.8665	0.19	19	0.5414	0.36	64%	28%	62651	29.83%	80.00%

Empty & Idle Initial Conditions

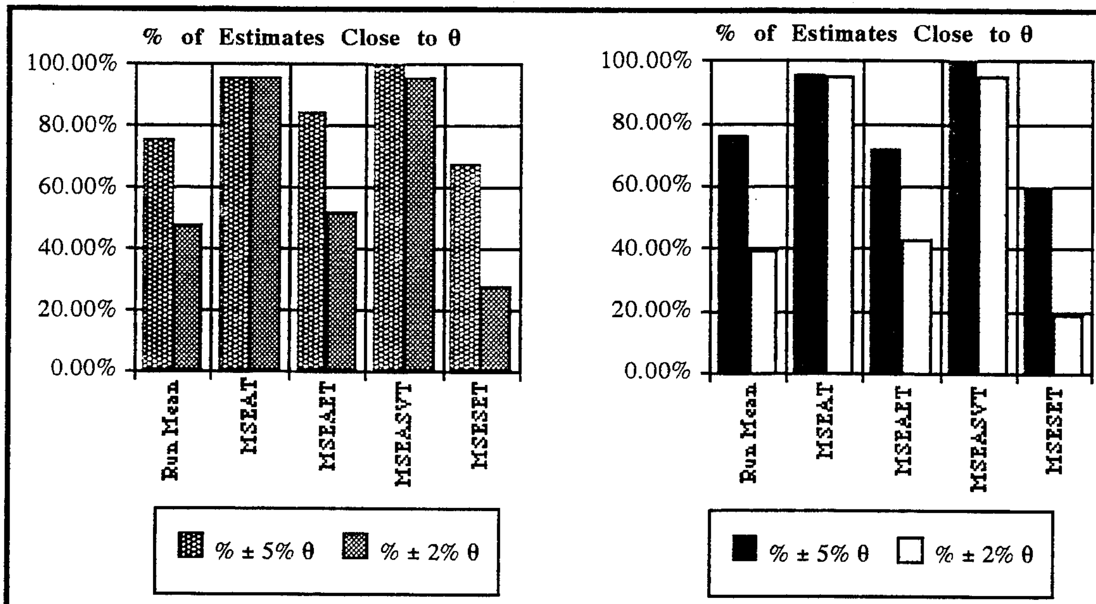
Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	0.9764	0.12	—	0.4672	0.29	76%	28%	.0	0.00%	100 %
MSEAT	2.5771	1.21	1	0.0700	0.06	100%	96%	156090	74.33%	100 %
MSEAET	1.6283	0.49	3	0.1001	0.13	100%	92%	145002	69.05%	100 %
MSEASVT	1.6348	0.78	5	0.0717	0.04	100%	100%	101881	48.51%	100 %
MSESET	0.8627	0.18	18	0.5833	0.35	56%	24%	59652	28.41%	88.00%

Table 4.4. Synopsis of $E_2/E_2/4$ (ρ = 0.90) Batch Means Experiments

4.2.2. U/Ln/3 Queueing Model

In general the U/Ln/3 queueing model batch mean experiments performed as did the $E_2/E_2/4$ queueing model ($\rho = 0.90$). We see an increase in the half-width, yet still provide accurate estimates of the expected wait time in the queue.

We do see in **Figure 4.7.** that this experiment that the MSEASVT method produces a better frequency of estimates close to the true than the MSEAT heuristic. This supports my contention that we can allow the output sequence to have a greater impact on the estimate while softening the power of the MSEAT.



U/Ln/3 (ρ = 0.90) w/Initial Conditions **U/Ln/3 (ρ = 0.90) Empty and Idle**

Figure 4.7. Results of U/Ln/3 (ρ = 0.98) Batch Means Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	0.1288	0.03	—	0.0666	0.06	76%	48%	0.00	0.00%	80.00%
MSEAT	0.2370	0.12	2	0.0265	0.003	96%	96%	108179	51.51%	100 %
MSEAET	0.3417	0.11	1	0.0667	0.05	84%	52%	165764	78.94%	100 %
MSEASVT	0.2021	0.09	3	0.0274	0.01	100%	96%	83248	39.64%	100 %
MSESET	0.1194	0.04	15	0.0951	0.07	68%	28%	67515	32.15%	80.00%

Empty & Idle Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	0.1325	0.02	—	0.0702	0.06	76%	40%	0.00	0.00%	80.00%
MSEAT	0.2809	0.16	2	0.0266	0.003	96%	96%	119927	57.11%	100 %
MSEAET	0.3518	0.13	1	0.0827	0.06	72%	44%	169780	80.85%	100 %
MSEASVT	0.2080	0.09	3	0.0273	0.01	100%	96%	87190	41.52%	100 %
MSESET	0.1209	0.03	14	0.1019	0.07	60%	20%	62136	29.59%	80.00%

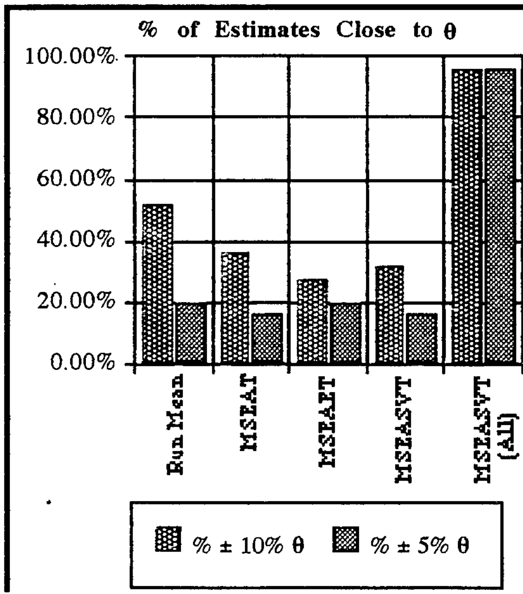
Table 4.5. Synopsis of U/Ln/3 (ρ = 0.90) Batch Means Experiments

4.2.3. $E_2/E_2/4$ Queueing Model (ρ = 0.98)

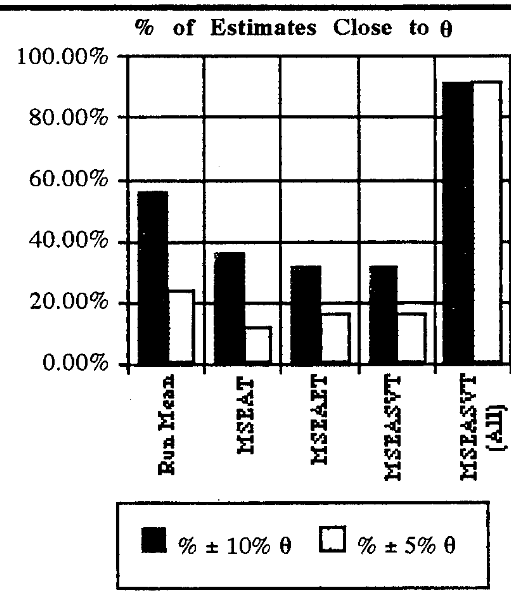
In this experiment, I allowed the heuristics to only seek a representative subsequence in the first half of the data for the truncation heuristics. This assumed that the second half-of the output sequence was more representative of the long-run distribution of the expected wait time in the queue. As I discussed earlier, I assumed a run length of 210,000 observations would be enough to travel through the transient phase. From **Figure 4.8.** we can see that this assumption failed.

I performed an additional analysis using the MSEASVT heuristic. In this analysis, I allowed the heuristic to search the entire output sequence as I had in previous batch means experiments. I did not use the MSESET heuristic because of its poor performance up to this point. In **Figure 4.8.**, the column "MSEASVT (All)" represents my results allowing the MSEASVT heuristic to search "all" of the output sequence.

Only the MSEASVT analysis across the entire data set produced an acceptable result. This intrigued me, since I knew that length of the transient phase is unique to the sample path. I wanted to see what run length it would take a sample path to travel through the transient phase. This is critical since the point that an apparent covariant stationary process begins is a random variable. No other research discusses the random nature of this point. I evaluate this in the next section.



$E_2/E_2/4$ (ρ = 0.98) w/Initial Conditions



$E_2/E_2/4$ (ρ = 0.98) Empty and Idle

Figure 4.8. Results of $E_2/E_2/4$ (ρ = 0.98) Batch Means Experiments

Stochastically Set Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	18.5432	7.94	—	10.9944	7.04	52%	20%	0.00	0.00%	72.00%
MSEAT	22.8314	11.76	5	12.7225	8.14	36%	16%	52983	25.23%	76.00%
MSEAET	18.0118	11.34	16	14.1276	9.52	28%	20%	64973	30.94%	56.00%
MSEASVT	22.2046	11.98	7	12.7693	8.13	32%	16%	44269	21.08%	80.00%
MSEASVT (All)	35.2173	15.88	2	0.8560	3.33	96%	96%	146188	69.61%	100 %

Empty & Idle Initial Conditions

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	d*	% Trunc	% Cvg
Run Mean	18.8812	7.91	—	10.8263	7.05	56%	24%	0.00	0.00%	72.00%
MSEAT	24.2643	11.77	4	12.9252	8.07	36%	12%	57933	27.59%	88.00%
MSEAET	18.7783	10.89	11	13.8823	8.43	32%	16%	64905	30.91%	60.00%
MSEASVT	22.0372	10.76	7	12.5352	8.23	32%	16%	42352	20.17%	92.00%
MSEASVT (All)	34.8337	15.31	2	1.3809	4.23	92%	92%	135933	64.73%	96.00%

Table 4.6. Synopsis of $E_2/E_2/4$ (ρ = 0.98) Batch Means Experiments

4.2.4. Analysis Across Batch Means Experiments

I did not focus my research on the length of the transient period. Rather I focused on the point estimate. Through my research, however, when I considered the batch means analysis I found, if you decide to use batch means as your primary method of creating independent observations for your estimator, you must consider the sensitivity of the estimate to a sample path run length.

Figure 4.9. shows a sample path for the empty and idle $E_2/E_2/4$ ($\rho = 0.98$) model. The straight line is the true expected value. We can that the process appears to settle nicely into a covariance stationary phase. Consider the Schruben test for initialization bias. It basically checks to see if the first half of an output sequence is similar to the second half. If it is, then Schruben's test concludes there is no initialization bias. If we were to truncate the output sequence at **"d*"**, we would find that the remaining output sequence would "fool" the Schruben test. As we can see, if we were to use a batch means analysis and be unfortunate enough to get this sample path when trying to estimate the true value of 92.796, we run the risk of making decisions based upon a biased estimate that appears to

approach its steady state value. This sample path is representative of those in my experiments.

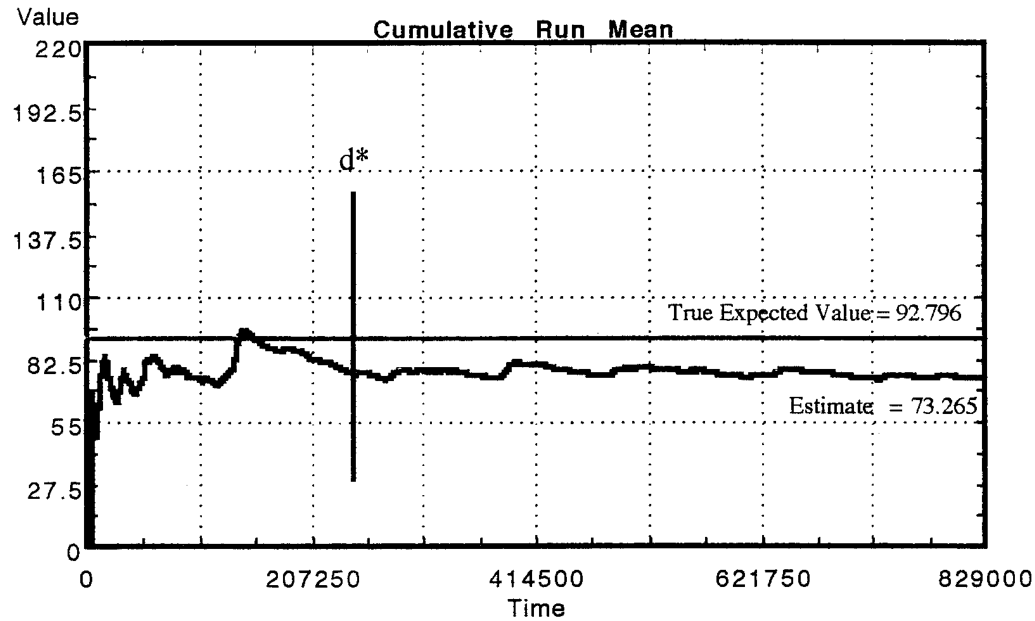


Figure 4.9. Batch Means Sample Path for 210,000 Observations

Figure 4.10. is a graphical depiction of the same sample path, but for a greater number of observations. As we can see, if we had used a longer run length, we would have seen an increase in the cumulative run mean. The chart shows d_1^*. This is the

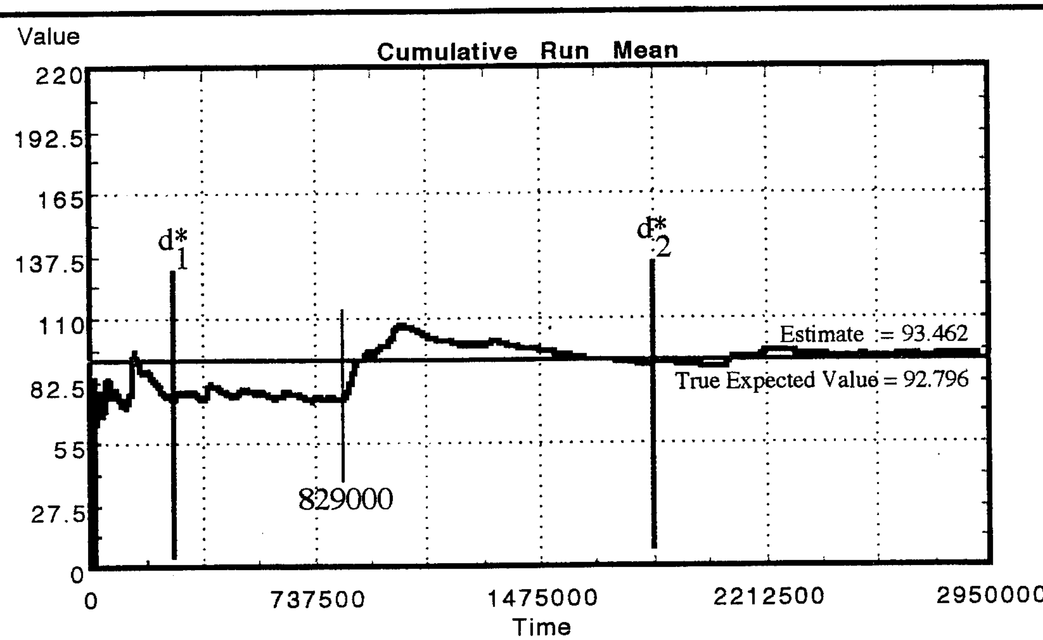


Figure 4.10. Batch Means Sample Path for 750,000 Observations

truncation point I showed in **Figure 4.9.** It is obvious that the end time of 829,000 does not complete the transient period for this sample path. Consider if we had used a run length of 280,000 observations, we would produce an estimate near 110. This emphasizes the point that we have no idea of the length of the transient period. The optimal truncation point is probably near d_2^*. This analysis furthers the concern that batch means methods can be very sensitive to simulation run length. Worse yet, we have no knowledge of these random characteristics.

Returning to the focus of my research, as in the replication deletion analysis, I provide a histogram of the bias observation for the batch means analysis. **Figure 4.11.** is misleading in that it appears that the Batch Mean estimates with no truncation produce frequently more estimates closer to the true expected wait time in the queue. **Figure 4.12.** is the histogram of the central bias observations. We can see from this chart that the MSEAT and MSEASVT heuristics still outperform the batch mean estimates by providing closer estimates to the true more frequently. The batch means estimator can only produce 41 out of 150 estimates within the interval -0.07 to 0.07, while the MSEAT and MSEASVT produce 96 and 95 out of 150 estimates within the same interval. This supports the replication/deletion conclusion that the heuristics provide more accurate estimates.

However, as I have shown above, the heuristics are not as precise as the sample mean estimate using the batch means methodology to create independent estimates. This presents interesting choices a decision maker must make prior to simulation. I would assume a decision maker would want the most accurate, and most precise estimate possible. If accuracy is the only criteria, then batch means performs as well, if not better, than replication deletion in providing an accurate point estimate.

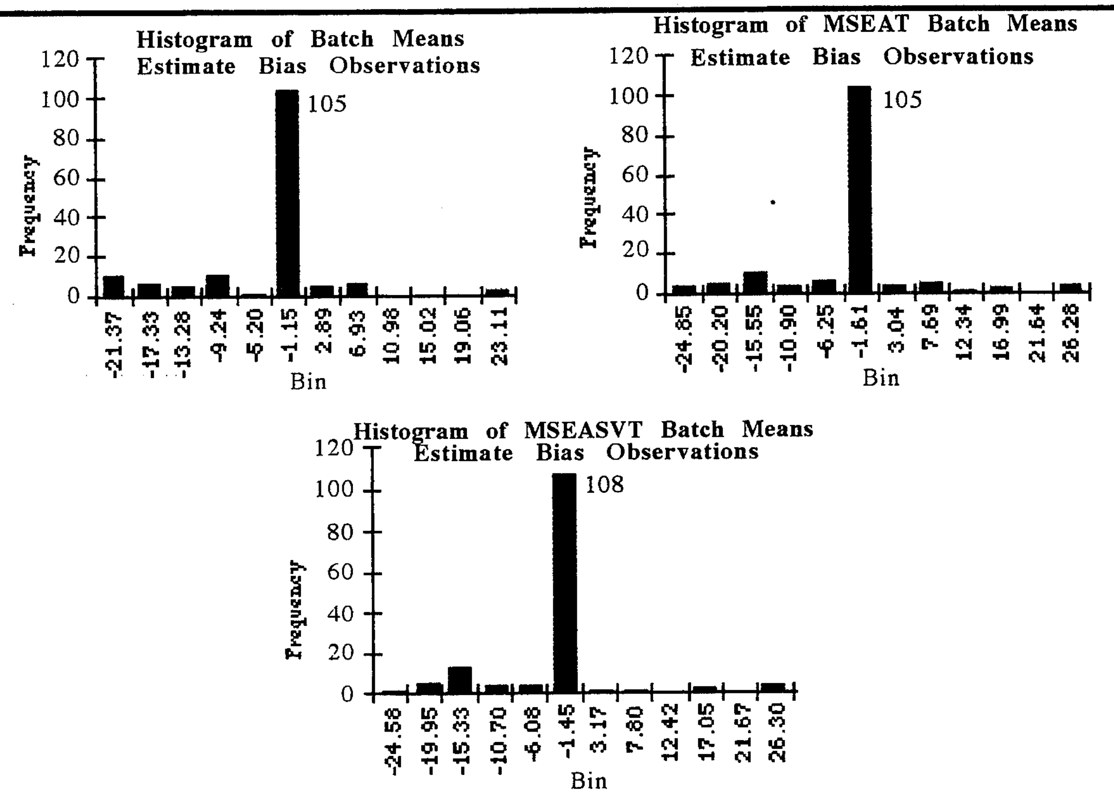


Figure 4.11. Histogram of Batch Means Bias Observations

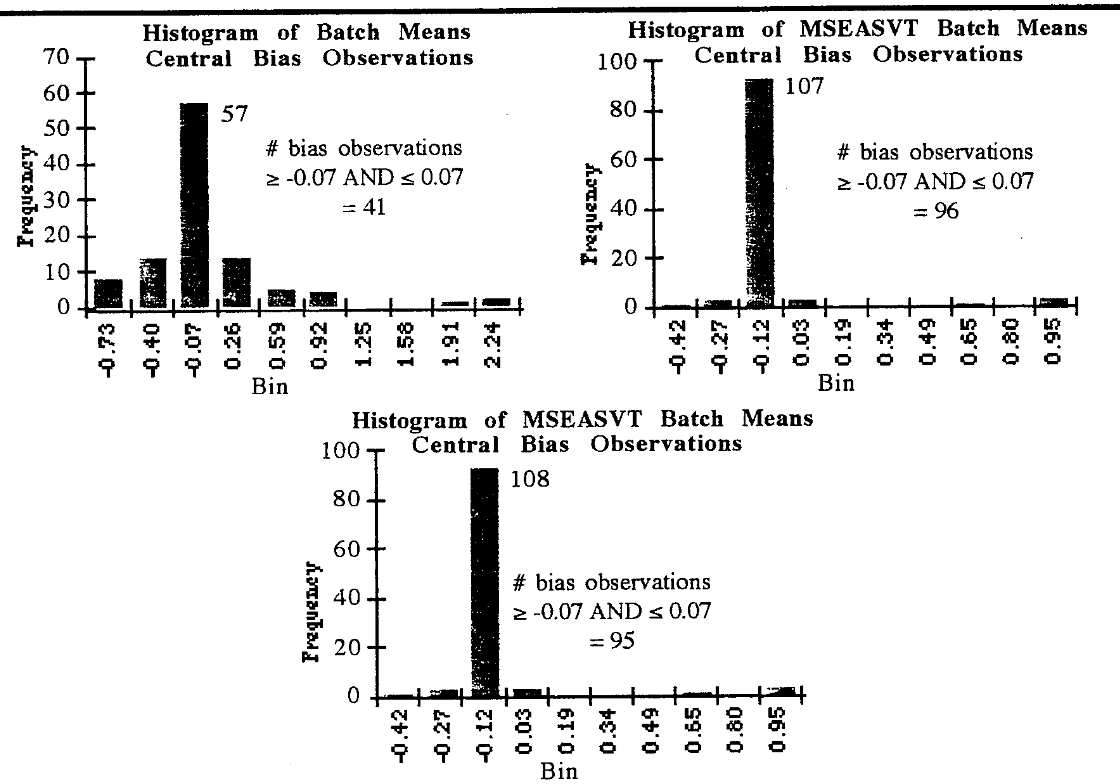


Figure 4.12. Histogram of Batch Means Central Bias Observations

4.3. M/M/2 & M/M/3 Tandem Queueing Model

As my final experiment, I performed my analysis across a tandem queueing model. **Figure 4.13.** supports all conclusions made thus far. The heuristics perform exceptionally in producing accurate and precise estimate. As **Table 4.7.** shows us, we reduced the half-width of estimate observation 25 out of 25 times when we compare the run mean half-width to those of the approximation heuristics. What is significant in **Table 4.7.** is that the heuristics provided as good a coverage as the run mean estimator. The fact that I continue to produce more accurate and more precise estimates than the run mean and equal the run mean coverage leads me to believe that the methodologies I provide in my research can only improve the performance of estimators used in more complex systems.

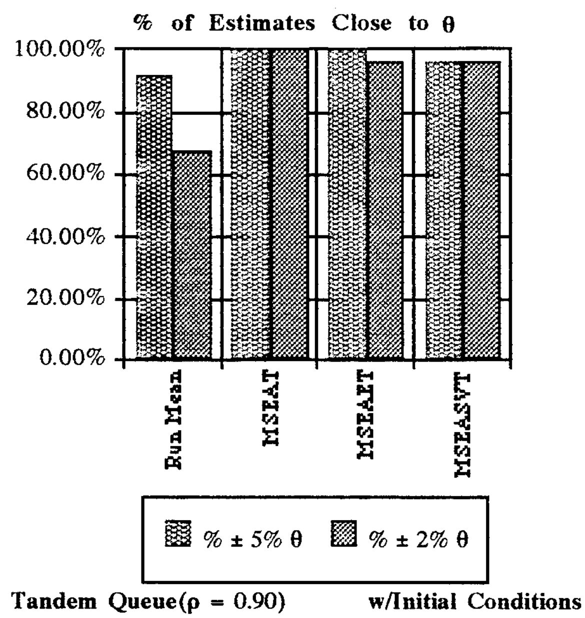


Figure 4.13. Results of M/M/2 and M/M3 Tandem Queue Experiments

Est. Method	Avg 1/2 W	Std Dev	# Decrease	Avg Bias	Std Dev	% ± 5% θ	% ± 2% θ	% Cvg
Run Mean	2.0032	0.48	—	0.7142	0.71	92.00%	68.00%	96%
MSEAT	0.2391	0.23	25	0.1273	0.13	100.00%	100.00%	96%
MSEAET	0.4881	0.23	25	0.2752	0.28	100.00%	96.00%	84%
MSEASVT	0.2885	0.24	25	0.3023	0.30	96.00%	96.00%	96%

Table 4.7. Synopsis of Tandem Queue (ρ = 0.90) Experiments

CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS

5.1. General

In this research, I developed four truncation heuristics of a simulation output sequence with the specific goal of producing a more accurate and more precise estimate than that of a standard output sequence mean.

We can see from Chapter 4. that using proven queueing approximations to stochastically set the initial conditions of the system reduces the initialization bias of a performance parameter estimate when we have a finite computer budget. Performing the stochastic initialization performs better for complex systems or systems with a large expected number in the system. This is true because a small expected number in the system has an associated shorter transient period. If this is the case, then starting the system stochastically shows statistically little improvement.

Although a finite run length may not get us through a transient period, the results lead me to believe that stochastically setting the initial conditions will reduce the length of the transient period. Additionally, using the approximations as the true mean value to perform a back-end truncation of output data at the point where the minimum estimated MSE occurs reduces the bias of the estimate. Even if an approximation had an absolute error of 20% to the true value, as shown, it could still produce a run estimate closer to the true expected value. Though the coverage across the 25 experiments for each model was not what I desired, you can easily see that the confidence half-width across each experiment was significantly smaller than that of the untruncated run mean data. This significant improvement in the precision is the reason for the shortage of coverage. Additionally, I consistently produced estimates closer to the true value of the mean. Thus, I achieved my goal of the research to produce a more precise and more accurate point estimate of the waiting time in a queue.

While I do acknowledge that more complex tests and further research be conducted on this topic, the preliminary findings of using the Whitt approximations to set initial conditions of a queueing simulation and assist us in getting and accurate an precise estimate are encouraging. I have found that my research supports Kelton's [17] contention that the methodology lessens the possibility of incorrect inferences from a data output sequence.

My truncation methods do lend themselves to a batch means analysis of one long run. However, the initial condition point of the one run negates the power of the steady state distribution approximation across multiple observations. I concede that at the beginning of each batch, a new initial "Queue Length" exists which should be more representative of steady state as the simulation run length increases. Since we do not know the length of each sample path's transient period, there is danger in using only one sample path to perform an analysis. I have shown this in my batch means experiments. Usually a person using batch means assumes his one long run will travel through a transient period. Since I assumed a finite computer budget, hedging one's bets on one sample path with no knowledge of the sample path's transient periods is risky. Using replication/deletion, we do not make any assumption other than we are most likely to not get through the transient phase. I have shown this more conservative approach to data analysis produces sound results.

The question still remains as to which is better; to have a coverage of estimates that could still have significant bias and variance, or less coverage with little or no bias and minimum variance? The decision maker is the only one who can answer this fundamental question. I submit, however, that since statistical analysis and inferences of output data can lead to errors based solely on the data, using the more precise and more accurate estimate should ultimately result in better outcome from decisions based upon that estimate.

5.2. Review of Contributions

Through my research, I have made considerable contributions. They are:

i) Provided insight into approximation-assisted control of initialization bias. This is a topic which has not received much research in the past.

ii) Developed a proven methodology which results in more accurate and precise estimates of a desired performance parameter than conventional methods.

iii) Eliminated the need for pilot runs to gain insight into the behavior of the system by maximizing the use of apriori information.

iv) Along with Dr. Manuel D. Rossetti, produced a paper on my research which may be published at the 1995 Winter Simulation Conference.

v) Produced a portable code and set of easily implementable algorithms for the methods presented herein. (See Appendix A for Algorithms)

vi) Provided a method that ultimately provides more accurate information possible in a decision making process than a sample mean.

5.3. Future Research

Since I focused on the estimated MSE as a truncation point heuristic, I assumed that there would be times when the bias reduction outweighed the variance increase to such a degree that I would have a half-width increase. While this was the case when I applied the heuristics in a batch means methodology, this was not the case for replication/deletion. The drastic reduction in the half-width makes an intriguing statement for possible variance reduction methods using approximations. Focusing on a data set and minimizing the variance with the assistance of an approximation may have significant application not only in queueing simulations, but simulations in general.

Other research areas that could continue from the groundwork established herein include:

i) Testing the heuristics on extremely complex models to evaluate their performance.

ii) Determining the length of the transient period prior to simulation using analytical approximations.

iii) Improve the half-width performance of the heuristics for batch means methods. One possibility is to batch the output sequence and allow the heuristics to search each batch for a sub-sequence that is more representative of the true expected value. If the output sequence had completely traversed the transient period we would expect the amount of data truncated per batch to decrease.

APPENDIX A. ALGORITHMS

A.1. Setting Up the Queueing Model Algorithm

Step 1. Determine number of parallel servers (m), arrival mean time ($1/\lambda$), service mean time ($1/\mu$), arrival variance (σ^2_a) and service variance (σ^2_s) in model design.

Step 2. Calculate Squared Coefficients of Variation for Arrival, Service and the System, and Traffic Intensity.

a. $c_a^2 = \dfrac{\sigma_a^2}{1/\lambda^2}$

b. $c_s^2 = \dfrac{\sigma_a^2}{1/\mu^2}$

c. $c^2 = \dfrac{c_a^2 + c_s^2}{2}$

d. $\rho = \dfrac{\lambda}{m\mu}$

Step 3. Calculate the Probability there are zero customers in system.

$$P_0 = \frac{1}{\left[\sum_{n=0}^{m-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^m}{m!} \frac{1}{1-\left(\frac{\lambda}{m\mu}\right)}\right]}$$

Step 4. Calculate the Probability "n" Customers are in the System for the M/M/m Queue

$$P_n = \begin{cases} \dfrac{(\lambda/\mu)^n P_0}{n!}, & if\ 0 \le n \le s \\ \dfrac{(\lambda/\mu)^n P_0}{m!m^{n-1}}, & if\ n > s \end{cases}$$

Step 5. Determine the probability of wait in the system $P[W(M/M/m)] > 0$

$$P[[W(M/M/m) > 0]] = [1 - P[W(M/M/m)] = 0] = \left[1 - \sum_{n=0}^{m-1} P_n\right]$$

Step 6. Using Whitt[37] Approximation, determine $\mathrm{P}(Q > 0)_{GI/G/m}$

a. $\mathrm{P}(Q > 0)_{GI/G/m} = \rho c^2 \mathrm{P}_{\mathrm{W(M/M/m)}}$

b. $\mathrm{P}(Q = 0)_{\mathrm{GI/G/m}} = 1 - \left(\rho c^2 \mathrm{P}_{\mathrm{W}_{(M/M/m)}}\right)$ **Probability there is no wait.

Step 7. Generate U~(0, 1)

a. If $U > P(Q > 0)_{GI/G/m}$, a queue exists.
   1. Generate $Q \sim P\{Q = k\}$.
   2. Return $N = Q + m$ **Stochastic Initial Condition**

b. If $U \leq P(Q > 0)_{GI/G/m}$, there is no queue. Must generate number in system
   1. Generate $N \sim p(k)$ **Truncated Poisson Distribution w/α Intensity**
   2. Return N. **Stochastic Initial Condition**

A.2. Expected Wait for the GI/G/m Queue Algorithm

(Used when $U \geq P(Q>0)_{GI/G/m}$)

Step 1. Calculate needed values for Whitt Approximations

a. $\gamma(m,\rho) = \min\{0.24, \dfrac{(1-\rho)(m-1)\left[(4+5m)^{\frac{1}{2}}-2\right]}{16m\rho}$

b. $\phi_1(m,\rho) = 1+\gamma(m,\rho)$

c. $\phi_2(m,\rho) = 1-4\gamma(m,\rho)$

d. $\phi_3(m,\rho) = \phi_2(m,\rho)\exp\left(\frac{-2(1-\rho)}{3\rho}\right)$

e. $\phi_4(m,\rho) = \min\left\{1, \dfrac{\phi_1(m,\rho)\ +\ \phi_3(m,\rho)}{2.0}\right\}$

f. $\Psi(c^2,m,\rho) = \begin{cases} 1 & c^2 \geq 1 \\ \phi_4(m,\rho)^{2(1-c^2)} & 0 \leq c^2 \leq 1 \end{cases}$

g. $\phi(\rho,c_a^2,c_s^2,) = \begin{cases} \left[\dfrac{4[c_a^2-c_s^2]}{4c_a^2-3c_s^2}\right]\phi_1(m,\rho) + \left[\dfrac{c_s^2}{4c_a^2-3c_s^2}\right]\Psi\left[\dfrac{c_a^2+c_s^2}{2},m,\rho\right] & c_a^2 \geq c_s^2 \\ \left[\dfrac{c_s^2-c_a^2}{2c_a^2+2c_s^2}\right]\phi_3(m,\rho) + \left[\dfrac{c_s^2+3c_a^2}{2c_a^2+2c_s^2}\right]\Psi\left[\dfrac{c_a^2+c_s^2}{2},m,\rho\right] & c_a^2 \leq c_s^2 \end{cases}$

h. Calculate the Expected Wait for the M/M/m Queue (= E[Queue Length]/λ)

$$E\left[W_{Q_{M/M/m}}\right] = \left[\frac{P_0\left(\frac{\lambda}{\tau}\right)^m}{(1-\rho)(m!)}\right]\frac{1}{\lambda}$$

i. Using Whit[37] Approximation, calculate the approximate Expected Wait in the Queue for the GI/G/m queue.

$$E\left[W_{Q_{GI/G/m}}\right] = \phi(\rho,c_a^2,c_s^2,m)\left[\frac{c_a^2+c_s^2}{2}\right]E\left[W_{Q_{M/M/m}}\right]$$

j. Calculate the Expected Conditional Queue Length GIVEN the Queue is NOT empty.

$$E\left[L_Q | L_Q > 0\right] = \lambda P\{Q>0\}E\left[W_{Q_{GI/G/m}}\right]$$

A.3. Whitt Approximation Case Decision Algorithm

Step 1. Calculate the Coefficients of Variation for Delay and the Conditional Queue Length.

a. $$d = \begin{cases} 3.0c_s^2(1.0 + c_s^2) & c_s^2 \geq 1.0 \\ (2.0c_s^2 + 1.0)(1.0 + c_s^2) & \text{otherwise} \end{cases}$$

b. $$c_d^2 = \frac{2\rho - 1 + (4(1-\rho)d)}{3(c_s^2 + 1)^2}$$

$$c_c^2 = \frac{1}{E[L_Q | L_Q > 0]} - 1.0 + (\rho(c_d^2 + 1))$$

Step 2. Determine Case Algorithm to use.

$$L_Q \text{ Case } = \begin{cases} \text{Case 1 if } c_c^2 > \left(1.0 - \left(\frac{1.0}{E[L_Q | L_Q > 0]}\right) + 0.02\right) \\ \text{Case 2 if } \left| c_c^2 - 1.0 + \left(\frac{1.0}{E[L_Q | L_Q > 0]}\right)\right| \leq 0.02 \\ \text{Case 3 if } c_c^2 > \frac{(E[L_Q | L_Q > 0])^2 - 1}{2(E[L_Q | L_Q > 0])^2} \text{ AND } c_c^2 \leq 1 - \frac{1}{(E[L_Q | L_Q > 0] - 0.02)} \\ \text{Case 4 if } c_c^2 \leq \frac{(E[L_Q | L_Q > 0])^2 - 1}{2(E[L_Q | L_Q > 0])^2} \end{cases}$$

A.4. Probability of Wait for GI/G/m Approximation Algorithm

Step 1. Calculate:

a. $$z = \frac{c_a^2 + c_s^2}{\left(1 + c_s^2\right)}$$

b. $$\gamma = \frac{(m - m\rho - 0.5)}{(m\rho z)^{\frac{1}{2}}}$$

c. $$\pi_4 = \min\left\{1, \frac{1 - \Phi\left[\left(1 + c_s^2\right)(1 - \rho)m^{1/2} / \left(c_a^2 + c_s^2\right)\right]}{1 - \Phi\left[(1 - \rho)m^{1/2}\right]}\left[P\left[W(M/M/m)\right] > 0\right]\right.$$

d. $$\pi_5 = \min\left\{1, \frac{1 - \Phi\left[2(1 - \rho)m^{1/2} / \left(1 + c_a^2\right)\right]}{1 - \Phi\left[(1 - \rho)m^{1/2}\right]}\left[P\left[W(M/M/m)\right] > 0\right]\right.$$

e. $$\pi_6 = 1 - \Phi\left[(m - m\rho - 0.5) \Big/ \sqrt{m\rho z}\right]$$

f. $$\pi_1 = \rho\pi_4 + (1 - \rho)^2 \pi_5$$

g. $$\pi_2 = c_a^2 \pi_1 + \left(1 - c_a^2\right)\pi_6$$

h. $$\pi_3 = 2\left(1 - c_a^2\right)(\gamma - 0.5)\pi_2 + \left[1 - \left[2\left(1 - c_a^2\right)(\gamma - 0.5)\right]\right]\pi_1$$

Step 2. Approximation for $P\{W_{GI/G/m} > 0\}$

$$P\{W_{GI/G/m} > 0\} = \min\{\pi, 1\}$$

$$\pi = \begin{cases} \pi_1 & \text{if } m \le 6 \text{ or } \gamma \le 0.5 \text{ or } c_a^2 \ge 1 \\ \pi_2 & \text{if } m \ge 7 \text{ and } \gamma \ge 1.0 \text{ and } c_a^2 < 1 \\ \pi_3 & \text{if } m \ge 7 \text{ and } c_a^2 < 1 \text{ and } 0.5 < \gamma < 1 \end{cases}$$

A.5. p(k) Truncated Poisson Distribution Algorithm

Step 1. Generate V~(0, 1)

Step 2. Calculate value of expected number of busy servers (L)

$$L = m(\rho - P\{Q = 0\})$$

Step 3. Generate value from Offered Load From Carried Load Function

α = OfferedLoadFromCarriedLoad(m, L)

Step 4. Set up an array of values such that

$$\text{Array}[i] = \frac{\alpha^i / i!}{\sum_{j=0}^{m+1} \frac{a^j}{j!}}$$

Step 5. Generate Truncated Poisson Distribution CDF Array for $0 \le i \le (m+1)$

TruncPoisCDF[0] = Array[0]

TruncPoisCDF[i] = TruncPoisCDF[i - 1] + Array[i]

Step 6. Return N = i such that V < Maximum TruncPoisCDF[i] Value. This is the stochastic initial number of customers in the system given there is not a queue.

A.6. OfferedLoadFromCarriedLoad(integer m, double L) Function Algorithm

Step 1. Set the maximum number of iterations (e.g.; iterations = 50)

Step 2. Set the error tolerance (e.g.; eps = 0.000001)

Step 3. Set initial starting point

$$\text{StartPoint} = L\left(\frac{\left(1.0 + \frac{L}{m}\right)}{m - L}\right)$$

Step 4. Perform Loop Calculation

$$\text{for}(\text{k} = 1;\ \text{i} \le \text{iterations};\ \text{k}++)$$

{

$$\text{b} = \text{ErlangFunc}(\text{m},\ \text{StartPoint})$$

$$\text{f} = (\text{StartPoint} * (1 - \text{b})) - \text{L}$$

$$\text{f1} = 1.0 - \text{b} - ((\text{m} - \text{StartPoint} + (\text{StartPoint} * \text{b})) * \text{b})$$

$$\text{if}(\text{f} < 1.0\text{e} - 10)$$

$$\text{f} = 0.0$$

$$\text{if}(\text{f1} < 1.0\text{e} - 10)$$

$$\text{f1} = 0.0$$

$$\text{a1} = \text{StartPoint} - \frac{\text{f}}{\text{f1}}$$

If the absolute value of a1—StartPoint < eps, StartPoint = a1.

}

Step 4. Return "a1" to the Truncate Poisson Distribution Function

A.7. ErlangFunc(integer c, double a) Algorithm

Step 1. Initialize real variables bn and b.

Step 2. Calculate $bn = \dfrac{a}{1.0 + a}$

Step 3. Perform Loop Calculation

$$
\begin{array}{l}
\text{for}(i = 2;\ i \le c;\ i{+}{+}) \\
\{ \\
\quad b = \dfrac{a * bn}{i + (a * bn)} \\
\quad bn = b \\
\}
\end{array}
$$

Step 4. Return "b" to Offered Load From Carried Load Function

A.8. Queue Length Generation Case 1 Algorithm (Mixture of Two Geometric Distributions)

Step 1. Calculate

a. $$\gamma = \frac{\left[1 + \sqrt{\left(1 - \frac{2}{\left[c^2 + 1 + \frac{1}{E\left[L_Q \mid L_Q > 0\right]}\right]}\right)}\right]}{2.0}$$

b. $$\text{m1} = \frac{E\left[L_Q \mid L_Q > 0\right]}{2.0\gamma}$$

c. $$\text{m2} = \frac{E\left[L_Q \mid L_Q > 0\right]}{2.0(1.0 - \gamma)}$$

d. $$\text{p1} = \frac{1.0}{\text{m1}}$$

e. $$\text{p2} = \frac{1.0}{\text{m2}}$$

Step 2. If p1 > 1.0, stop and proceed to Case 2.

Step 3. If p1 ≤ 1.0, generate R~(0, 1)

Step 4. If R ≤ γ Set Initial Number in the System to $N = 1.0 + \left\lceil \frac{\ln(1-\text{R})}{\ln(1-\text{p1})} \right\rceil + \text{m}$

Step 5. If R > γ Set Initial Number in the System to $N = 1.0 + \left\lceil \frac{\ln(1-\text{R})}{\ln(1-\text{p2})} \right\rceil + \text{m}$

A.9. Queue Length Generation Case 2 Algorithm (Simple Geometric Distributions)

Step 1. Calculate $p1 = \frac{1.0}{E\left[L_Q | L_Q > 0\right]} p1$

Step 2. Generate R~(0, 1).

Step 3. Set Initial Number in the System to $N = 1.0 + \left\lceil \frac{\ln(1-R)}{\ln(1-p1)} \right\rceil + m$

A.10. Queue Length Generation Case 3 Algorithm (Convolution of Two Geometric Distributions)

Step 1. Calculate

a.
$$m_1 = \frac{\left(\left(E[L_Q|L_Q>0]-1\right)-\left[\left(E[L_Q|L_Q>0]-1\right)^2\right]^{1/2}\right)}{2} - \frac{2\left(E[L_Q|L_Q>0]\right)^2\left(1-c_C^2-\frac{1}{E[L_Q|L_Q>0]}\right)}{2}$$

b.
$$m_2 = \frac{\left(\left(E[L_Q|L_Q>0]+1\right)+\left[\left(E[L_Q|L_Q>0]-1\right)^2\right]^{1/2}\right)}{2} - \frac{2\left(E[L_Q|L_Q>0]\right)^2\left(1-c_C^2-\frac{1}{E[L_Q|L_Q>0]}\right)}{2}$$

c. $p_1 = \frac{1}{m_1+1}$ $\qquad p_2 = \frac{1}{m_2}$

Step 2. Generate U~(0, 1). $Y_1 = \frac{ln\,U}{ln(1-p_1)}$

Step 3. Generate U~(0, 1). $Y_2 = \frac{ln\,U}{ln(1-p_2)}$

Step 4. Return N = $Y_1 + Y_2 + m$

A.11. Queue Length Generation Case 4 Algorithm (Convolution of Two Geometric Distributions)

Step 1. Calculate

a. $m_1 = \dfrac{E[L_Q | L_Q > 0] - 1}{2}$

b. $m_2 = \dfrac{E[L_Q | L_Q > 0] + 1}{2}$

c. $p_1 = \dfrac{1}{m_1 + 1}$ $\quad p_2 = \dfrac{1}{m_2}$

Step 2. Generate U~(0, 1). $Y_1 = \dfrac{ln U}{ln(1 - p_1)}$

Step 3. Generate U~(0, 1). $Y_2 = \dfrac{ln U}{ln(1 - p_2)}$

Step 4. Return N = $Y_1 + Y_2 + m$

APPENNDIX B. EXPERIMENT RESULTS DATA

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.9938	—	0.1660	1	1	0	0.00%	13.60596
	MSEAT	0.5549	1	-0.1345	1	1	6113	61.13%	13.30546
	MSEAET	0.5552	1	-0.1354	1	1	5602	56.02%	13.30464
	MSEASVT	0.5661	1	-0.2092	1	1	3850	38.50%	13.23082
	MSESET	1.1796	0	-0.2740	1	0	3452	34.52%	13.16604
2	Run Mean	0.8289	—	-0.3299	1	0	0	0.00%	13.11010
	MSEAT	0.2846	1	-0.2676	1	1	6723	67.23%	13.17245
	MSEAET	0.2862	1	-0.2710	1	0	6709	67.09%	13.16898
	MSEASVT	0.2925	1	-0.3061	1	0	5117	51.17%	13.13386
	MSESET	1.1266	0	0.3525	1	0	3447	34.47%	13.79251
3	Run Mean	1.0613	—	-0.4181	1	0	0	0.00%	13.02190
	MSEAT	0.0994	1	-0.1508	1	1	6716	67.16%	13.28918
	MSEAET	0.1039	1	-0.1592	1	1	6373	63.73%	13.28083
	MSEASVT	0.1061	1	-0.1311	1	1	4805	48.05%	13.30891
	MSESET	1.7295	0	0.2331	1	1	3249	32.49%	13.67312
4	Run Mean	0.6825	—	0.0220	1	1	0	0.00%	13.46201
	MSEAT	0.1654	1	-0.1256	1	1	4833	48.33%	13.31436
	MSEAET	0.1696	1	-0.1251	1	1	5110	51.10%	13.31491
	MSEASVT	0.1772	1	-0.1672	1	1	3551	35.51%	13.27283
	MSESET	0.8102	0	-0.5016	1	0	3146	31.46%	12.93841
5	Run Mean	0.9510	—	0.6470	1	0	0	0.00%	14.08703
	MSEAT	0.3045	1	0.1076	1	1	6079	60.79%	13.54757
	MSEAET	0.3044	1	0.1028	1	1	5619	56.19%	13.54282
	MSEASVT	0.3117	1	0.1262	1	1	4304	43.04%	13.56621
	MSESET	1.3343	0	0.5421	1	0	3119	31.19%	13.98211
6	Run Mean	0.6437	—	0.0235	1	1	0	0.00%	13.46355
	MSEAT	0.1357	1	-0.0009	1	1	7102	71.02%	13.43911
	MSEAET	0.1428	1	-0.0228	1	1	6952	69.52%	13.41717
	MSEASVT	0.1467	1	-0.0172	1	1	4674	46.74%	13.42277
	MSESET	0.9767	0	0.3537	1	0	3326	33.26%	13.79370
7	Run Mean	0.8717	—	0.0735	1	1	0	0.00%	13.51350
	MSEAT	0.4269	1	-0.0515	1	1	6366	63.66%	13.38846
	MSEAET	0.4270	1	-0.0531	1	1	6248	62.48%	13.38693
	MSEASVT	0.4314	1	-0.0667	1	1	4850	48.50%	13.37330
	MSESET	1.3434	0	0.6057	1	0	3170	31.70%	14.04568
8	Run Mean	0.5926	—	-0.6544	0	0	0	0.00%	12.78565
	MSEAT	0.0812	1	-0.1231	1	1	5606	56.06%	13.31692
	MSEAET	0.0803	1	-0.1339	1	1	5479	54.79%	13.30610
	MSEASVT	0.0920	1	-0.1712	1	1	3926	39.26%	13.26878
	MSESET	0.9348	0	-0.9010	0	0	3410	34.10%	12.53898
9	Run Mean	0.7426	—	-0.0172	1	1	0	0.00%	13.42280
	MSEAT	0.3224	1	0.0480	1	1	5174	51.74%	13.48800
	MSEAET	0.3243	1	0.0287	1	1	5123	51.23%	13.46866
	MSEASVT	0.3227	1	0.0437	1	1	4001	40.01%	13.48367
	MSESET	1.2074	0	-0.2266	1	1	3539	35.39%	13.21337

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.6425	—	-0.2207	1	1	0	0.00%	13.21933
	MSEAT	0.2789	1	-0.0166	1	1	5943	59.43%	13.42336
	MSEAET	0.2793	1	-0.0241	1	1	5934	59.34%	13.41595
	MSEASVT	0.2881	1	-0.0286	1	1	5442	54.42%	13.41136
	MSESET	1.0271	0	0.0062	1	1	3374	33.74%	13.44615
11	Run Mean	0.9582	—	-0.0130	1	1	0	0.00%	13.42703
	MSEAT	0.2216	1	0.1171	1	1	6032	60.32%	13.55714
	MSEAET	0.2255	1	0.0986	1	1	5959	59.59%	13.53862
	MSEASVT	0.2250	1	0.1089	1	1	4998	49.98%	13.54894
	MSESET	1.2604	0	0.4162	1	0	3001	30.01%	13.85621
12	Run Mean	0.7612	—	-0.0531	1	1	0	0.00%	13.38688
	MSEAT	0.2236	1	-0.2027	1	1	5745	57.45%	13.23730
	MSEAET	0.2237	1	-0.2022	1	1	5983	59.83%	13.23785
	MSEASVT	0.2281	1	-0.2346	1	1	4404	44.04%	13.20542
	MSESET	0.9489	0	-0.5205	1	0	3392	33.92%	12.91949
13	Run Mean	0.6297	—	-0.4586	1	0	0	0.00%	12.98142
	MSEAT	0.3135	1	0.0753	1	1	6141	61.41%	13.51526
	MSEAET	0.3139	1	0.0711	1	1	5737	57.37%	13.51115
	MSEASVT	0.3235	1	0.0448	1	1	3533	35.33%	13.48476
	MSESET	1.2982	0	-0.6239	1	0	3480	34.80%	12.81614
14	Run Mean	0.5948	—	-0.4203	1	0	0	0.00%	13.01965
	MSEAT	0.2410	1	-0.0438	1	1	5359	53.59%	13.39621
	MSEAET	0.2413	1	-0.0409	1	1	4860	48.60%	13.39910
	MSEASVT	0.2435	1	-0.0659	1	1	4099	40.99%	13.37413
	MSESET	0.8984	0	-0.4204	1	0	3515	35.15%	13.01959
15	Run Mean	0.7944	—	-0.0908	1	1	0	0.00%	13.34922
	MSEAT	0.1998	1	-0.2056	1	1	6324	63.24%	13.23437
	MSEAET	0.1998	1	-0.2298	1	1	6158	61.58%	13.21024
	MSEASVT	0.2053	1	-0.1907	1	1	4481	44.81%	13.24927
	MSESET	0.8831	0	-0.1234	1	1	3274	32.74%	13.31664
16	Run Mean	0.6298	—	-0.5943	1	0	0	0.00%	12.84573
	MSEAT	0.2485	1	-0.1163	1	1	6243	62.43%	13.32375
	MSEAET	0.2483	1	-0.1276	1	1	5619	56.19%	13.31241
	MSEASVT	0.2534	1	-0.1354	1	1	4488	44.88%	13.30462
	MSESET	0.8478	0	-1.0690	0	0	2725	27.25%	12.37097
17	Run Mean	1.0174	—	0.1323	1	1	0	0.00%	13.57226
	MSEAT	0.0409	1	-0.0299	1	1	6454	64.54%	13.41008
	MSEAET	0.0258	1	-0.0651	1	1	6216	62.16%	13.37486
	MSEASVT	0.0672	1	-0.0101	1	1	5110	51.10%	13.42992
	MSESET	1.0158	1	0.0773	1	1	3448	34.48%	13.51726
18	Run Mean	0.5785	—	0.2080	1	1	0	0.00%	13.64800
	MSEAT	0.0403	1	-0.0645	1	1	5209	52.09%	13.37555
	MSEAET	0.0411	1	-0.0855	1	1	5451	54.51%	13.35455
	MSEASVT	0.0637	1	-0.0395	1	1	3681	36.81%	13.40050
	MSESET	0.9808	0	0.2047	1	1	3174	31.74%	13.64466

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	0.8325	—	-0.1512	1	1	0	0.00%	13.28876
	MSEAT	0.0546	1	-0.0385	1	1	6423	64.23%	13.40154
	MSEAET	0.0517	1	-0.0625	1	1	6343	63.43%	13.37754
	MSEASVT	0.0595	1	-0.0412	1	1	4917	49.17%	13.39880
	MSESET	1.4523	0	0.4474	1	0	3129	31.29%	13.88742
20	Run Mean	0.7180	—	-0.0898	1	1	0	0.00%	13.35020
	MSEAT	0.1228	1	-0.1160	1	1	6726	67.26%	13.32401
	MSEAET	0.1232	1	-0.1292	1	1	6284	62.84%	13.31080
	MSEASVT	0.1322	1	-0.1180	1	1	4982	49.82%	13.32198
	MSESET	0.9836	0	-0.0619	1	1	3885	38.85%	13.37808
21	Run Mean	1.0256	—	0.6534	1	0	0	0.00%	14.09338
	MSEAT	0.2364	1	0.0616	1	1	6574	65.74%	13.50162
	MSEAET	0.2373	1	0.0778	1	1	6270	62.70%	13.51784
	MSEASVT	0.2509	1	0.1248	1	1	4503	45.03%	13.56480
	MSESET	1.0646	0	0.4714	1	0	3409	34.09%	13.91141
22	Run Mean	0.7057	—	-0.1918	1	1	0	0.00%	13.24822
	MSEAT	0.2456	1	-0.2978	1	0	5167	51.67%	13.14225
	MSEAET	0.2467	1	-0.2958	1	0	5541	55.41%	13.14420
	MSEASVT	0.2511	1	-0.3493	1	0	3820	38.20%	13.09069
	MSESET	1.0823	0	-0.6284	1	0	3110	31.10%	12.81157
23	Run Mean	0.9242	—	-0.1515	1	1	0	0.00%	13.28854
	MSEAT	0.2578	1	-0.2567	1	1	6467	64.67%	13.18327
	MSEAET	0.2595	1	-0.2587	1	1	6433	64.33%	13.18131
	MSEASVT	0.2598	1	-0.2627	1	1	5512	55.12%	13.17727
	MSESET	1.2103	0	-0.1016	1	1	3241	32.41%	13.33842
24	Run Mean	1.0277	—	0.7468	0	0	0	0.00%	14.18676
	MSEAT	0.1885	1	-0.1016	1	1	6107	61.07%	13.33840
	MSEAET	0.1890	1	-0.0953	1	1	6111	61.11%	13.34469
	MSEASVT	0.2223	1	-0.1357	1	1	4462	44.62%	13.30432
	MSESET	1.0333	0	0.9205	0	0	3532	35.32%	14.36053
25	Run Mean	0.8325	—	-0.1512	1	1	0	0.00%	13.28876
	MSEAT	0.0546	1	-0.0385	1	1	6423	64.23%	13.40154
	MSEAET	0.0517	1	-0.0625	1	1	6343	63.43%	13.37754
	MSEASVT	0.0595	1	-0.0412	1	1	4917	49.17%	13.39880
	MSESET	1.4523	0	0.4474	1	0	3129	31.29%	13.88742

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.8642	—	0.1174	1	1	0	0.00%	13.55743
	MSEAT	0.2460	1	-0.2047	1	1	5928	59.28%	13.23535
	MSEAET	0.2479	1	-0.2084	1	1	5912	59.12%	13.23162
	MSEASVT	0.2659	1	-0.2409	1	1	4203	42.03%	13.19915
	MSESET	1.1719	0	-0.0576	1	1	3282	32.82%	13.38240
2	Run Mean	0.8372	—	-0.3110	1	0	0	0.00%	13.12897
	MSEAT	0.0606	1	-0.1110	1	1	7346	73.46%	13.32901
	MSEAET	0.0680	1	-0.1111	1	1	7325	73.25%	13.32892
	MSEASVT	0.0999	1	-0.1189	1	1	5859	58.59%	13.32112
	MSESET	1.1501	0	0.3097	1	0	3576	35.76%	13.74965
3	Run Mean	1.1142	—	-0.2797	1	0	0	0.00%	13.16034
	MSEAT	0.1309	1	-0.1508	1	1	6673	66.73%	13.28923
	MSEAET	0.1334	1	-0.1734	1	1	6293	62.93%	13.26661
	MSEASVT	0.1389	1	-0.1557	1	1	4787	47.87%	13.28427
	MSESET	1.8350	0	0.5551	1	0	3450	34.50%	13.99507
4	Run Mean	0.6932	—	-0.0322	1	1	0	0.00%	13.40781
	MSEAT	0.0607	1	-0.0898	1	1	5277	52.77%	13.35020
	MSEAET	0.0663	1	-0.0817	1	1	5349	53.49%	13.35831
	MSEASVT	0.0722	1	-0.0967	1	1	4504	45.04%	13.34330
	MSESET	0.7859	0	-0.5377	1	0	2933	29.33%	12.90230
5	Run Mean	0.8144	—	0.4228	1	0	0	0.00%	13.86282
	MSEAT	0.2717	1	0.0278	1	1	6403	64.03%	13.46784
	MSEAET	0.2721	1	0.0230	1	1	5844	58.44%	13.46295
	MSEASVT	0.2775	1	0.0107	1	1	4231	42.31%	13.45073
	MSESET	1.1773	0	0.2407	1	1	3174	31.74%	13.68067
6	Run Mean	0.6979	—	0.0170	1	1	0	0.00%	13.45696
	MSEAT	0.2042	1	-0.0877	1	1	6896	68.96%	13.35227
	MSEAET	0.2090	1	-0.0808	1	1	6444	64.44%	13.35920
	MSEASVT	0.2135	1	-0.1073	1	1	4549	45.49%	13.33275
	MSESET	0.9996	0	0.5324	1	0	3225	32.25%	13.97240
7	Run Mean	0.8929	—	0.1333	1	1	0	0.00%	13.57335
	MSEAT	0.4653	1	-0.0849	1	1	6063	60.63%	13.35515
	MSEAET	0.4654	1	-0.0928	1	1	6025	60.25%	13.34725
	MSEASVT	0.4862	1	-0.1059	1	1	4496	44.96%	13.33405
	MSESET	1.4108	0	0.6719	1	0	3344	33.44%	14.11186
8	Run Mean	0.5962	—	-0.6416	1	0	0	0.00%	12.79845
	MSEAT	0.0789	1	-0.1123	1	1	5484	54.84%	13.32774
	MSEAET	0.0813	1	-0.1351	1	1	5452	54.52%	13.30485
	MSEASVT	0.0983	1	-0.1807	1	1	3989	39.89%	13.25928
	MSESET	0.8554	0	-0.9434	0	0	3460	34.60%	12.49660
9	Run Mean	0.8140	—	-0.1625	1	1	0	0.00%	13.27752
	MSEAT	0.2293	1	0.1056	1	1	5410	54.10%	13.54562
	MSEAET	0.2235	1	0.1033	1	1	5724	57.24%	13.54329
	MSEASVT	0.2527	1	0.0832	1	1	4011	40.11%	13.52324
	MSESET	1.0977	0	-0.3182	1	0	3708	37.08%	13.12177

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.6559	—	-0.2322	1	1	0	0.00%	13.20780
	MSEAT	0.2536	1	-0.0105	1	1	6358	63.58%	13.42947
	MSEAET	0.2543	1	-0.0187	1	1	6153	61.53%	13.42134
	MSEASVT	0.2556	1	-0.0325	1	1	5267	52.67%	13.40751
	MSESET	1.0683	0	0.0688	1	1	3361	33.61%	13.50878
11	Run Mean	0.9607	—	0.0323	1	1	0	0.00%	13.47227
	MSEAT	0.2518	1	0.0332	1	1	6332	63.32%	13.47320
	MSEAET	0.2527	1	0.0265	1	1	6469	64.69%	13.46647
	MSEASVT	0.2566	1	-0.0003	1	1	5062	50.62%	13.43973
	MSESET	1.2228	0	0.6294	1	0	3110	31.10%	14.06945
12	Run Mean	0.6695	—	-0.1390	1	1	0	0.00%	13.30097
	MSEAT	0.1671	1	-0.1166	1	1	6474	64.74%	13.32337
	MSEAET	0.1682	1	-0.1178	1	1	6264	62.64%	13.32218
	MSEASVT	0.1791	1	-0.1714	1	1	5357	53.57%	13.26858
	MSESET	1.0718	0	-0.0130	1	1	3298	32.98%	13.42699
13	Run Mean	0.7040	—	-0.3175	1	0	0	0.00%	13.12250
	MSEAT	0.2420	1	0.0659	1	1	6385	63.85%	13.50586
	MSEAET	0.2434	1	0.0538	1	1	5741	57.41%	13.49385
	MSEASVT	0.2460	1	0.0664	1	1	4132	41.32%	13.50640
	MSESET	1.3877	0	-0.2751	1	0	3425	34.25%	13.16485
14	Run Mean	0.5994	—	-0.5437	1	0	0	0.00%	12.89633
	MSEAT	0.2443	1	-0.0520	1	1	6221	62.21%	13.38796
	MSEAET	0.2445	1	-0.0588	1	1	5692	56.92%	13.38122
	MSEASVT	0.2463	1	-0.0557	1	1	4702	47.02%	13.38432
	MSESET	0.8853	0	-0.6667	0	0	3309	33.09%	12.77332
15	Run Mean	0.8581	—	-0.1527	1	1	0	0.00%	13.28725
	MSEAT	0.1931	1	-0.2446	1	1	5692	56.92%	13.19540
	MSEAET	0.1921	1	-0.2533	1	1	5818	58.18%	13.18674
	MSEASVT	0.2048	1	-0.2305	1	1	4686	46.86%	13.20949
	MSESET	1.0573	0	-0.1020	1	1	3178	31.78%	13.33798
16	Run Mean	0.6616	—	-0.5655	1	0	0	0.00%	12.87454
	MSEAT	0.3784	1	-0.0346	1	1	5011	50.11%	13.40541
	MSEAET	0.3799	1	-0.0384	1	1	4511	45.11%	13.40160
	MSEASVT	0.3789	1	-0.0449	1	1	3855	38.55%	13.39513
	MSESET	0.9547	0	-1.0035	0	0	3025	30.25%	12.43645
17	Run Mean	1.0375	—	0.1463	1	1	0	0.00%	13.58632
	MSEAT	0.1604	1	-0.0072	1	1	6854	68.54%	13.43276
	MSEAET	0.1609	1	-0.0146	1	1	6790	67.90%	13.42542
	MSEASVT	0.1685	1	-0.0361	1	1	5045	50.45%	13.40389
	MSESET	1.1561	0	0.1196	1	1	3579	35.79%	13.55960
18	Run Mean	0.6128	—	0.1291	1	1	0	0.00%	13.56912
	MSEAT	0.0061	1	-0.0616	1	1	6371	63.71%	13.37841
	MSEAET	0.0018	1	-0.0581	1	1	6578	65.78%	13.38188
	MSEASVT	0.0497	1	-0.0208	1	1	3856	38.56%	13.41916
	MSESET	0.8990	0	0.0584	1	1	2887	28.87%	13.49839

Micro Analysis of Results

Model: E_2 / E_2 / 4 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	0.7925	—	-0.1890	1	1	0	0.00%	13.25097
	MSEAT	0.0427	1	-0.0536	1	1	5618	56.18%	13.38644
	MSEAET	0.0546	1	-0.0555	1	1	5828	58.28%	13.38447
	MSEASVT	0.0558	1	-0.0927	1	1	4626	46.26%	13.34728
	MSESET	1.3894	0	0.3649	1	0	3354	33.54%	13.80492
20	Run Mean	0.6945	—	-0.1121	1	1	0	0.00%	13.32791
	MSEAT	0.1241	1	-0.1102	1	1	6684	66.84%	13.32985
	MSEAET	0.1228	1	-0.1294	1	1	6562	65.62%	13.31062
	MSEASVT	0.1312	1	-0.1447	1	1	4339	43.39%	13.29532
	MSESET	1.0446	0	-0.1063	1	1	3777	37.77%	13.33366
21	Run Mean	1.0031	—	0.3688	1	0	0	0.00%	13.80878
	MSEAT	0.2454	1	0.0250	1	1	6133	61.33%	13.46497
	MSEAET	0.2461	1	0.0141	1	1	6298	62.98%	13.45408
	MSEASVT	0.2492	1	0.0151	1	1	3878	38.78%	13.45513
	MSESET	1.0545	0	0.0549	1	1	3202	32.02%	13.49490
22	Run Mean	0.7707	—	-0.1529	1	1	0	0.00%	13.28712
	MSEAT	0.2272	1	-0.2893	1	0	5435	54.35%	13.15071
	MSEAET	0.2285	1	-0.2887	1	0	5589	55.89%	13.15134
	MSEASVT	0.2253	1	-0.3094	1	0	3990	39.90%	13.13056
	MSESET	1.0477	0	-0.7285	0	0	3187	31.87%	12.71154
23	Run Mean	0.8687	—	-0.3716	1	0	0	0.00%	13.06838
	MSEAT	0.1705	1	-0.2257	1	1	6313	63.13%	13.21431
	MSEAET	0.1718	1	-0.2144	1	1	6684	66.84%	13.22556
	MSEASVT	0.1844	1	-0.2018	1	1	4914	49.14%	13.23821
	MSESET	1.1242	0	-0.3963	1	0	3252	32.52%	13.04368
24	Run Mean	1.0175	—	0.6646	1	0	0	0.00%	14.10464
	MSEAT	0.0832	1	-0.0227	1	1	6822	68.22%	13.41732
	MSEAET	0.0826	1	0.0153	1	1	6429	64.29%	13.45526
	MSEASVT	0.0897	1	0.0010	1	1	4164	41.64%	13.44103
	MSESET	1.1041	0	0.8354	0	0	3792	37.92%	14.27544
25	Run Mean	0.7925	—	-0.1890	1	1	0	0.00%	13.25097
	MSEAT	0.0427	1	-0.0536	1	1	5618	56.18%	13.38644
	MSEAET	0.0546	1	-0.0555	1	1	5828	58.28%	13.38447
	MSEASVT	0.0558	1	-0.0927	1	1	4626	46.26%	13.34728
	MSESET	1.3894	0	0.3649	1	0	3354	33.54%	13.80492

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.1549	—	0.0989	1	0	0	0.00%	2.35991
	MSEAT	0.0021	1	0.0275	1	1	18534	61.78%	2.28851
	MSEAET	0.0283	1	0.0066	1	1	21868	72.89%	2.26761
	MSEASVT	0.0093	1	0.0209	1	1	15796	52.65%	2.28191
	MSESET	0.2310	0	0.0935	1	0	8196	27.32%	2.35451
2	Run Mean	0.1586	—	0.1158	0	0	0	0.00%	2.37680
	MSEAT	0.0004	1	0.0266	1	1	22097	73.66%	2.28760
	MSEAET	0.0422	1	0.0245	1	1	24166	80.55%	2.28551
	MSEASVT	0.0089	1	0.0234	1	1	14391	47.97%	2.28443
	MSESET	0.1553	1	0.0980	1	0	8228	27.43%	2.35899
3	Run Mean	0.1673	—	0.0451	1	1	0	0.00%	2.30609
	MSEAT	0.0709	1	0.0623	1	0	20573	68.58%	2.32328
	MSEAET	0.0873	1	0.0633	1	0	22706	75.69%	2.32435
	MSEASVT	0.0742	1	0.0626	1	0	13111	43.70%	2.32363
	MSESET	0.1755	0	0.0179	1	1	10264	34.21%	2.27891
4	Run Mean	0.1077	—	0.0579	1	0	0	0.00%	2.31891
	MSEAT	0.0157	1	0.0188	1	1	16997	56.66%	2.27978
	MSEAET	0.0407	1	0.0074	1	1	21942	73.14%	2.26841
	MSEASVT	0.0157	1	0.0156	1	1	12799	42.66%	2.27659
	MSESET	0.0785	1	-0.0221	1	1	6378	21.26%	2.23892
5	Run Mean	0.0963	—	0.0600	1	0	0	0.00%	2.32100
	MSEAT	0.0001	1	0.0265	1	1	17187	57.29%	2.28751
	MSEAET	0.0288	1	0.0349	1	1	21416	71.39%	2.29593
	MSEASVT	0.0010	1	0.0269	1	1	11599	38.66%	2.28787
	MSESET	0.1351	0	0.1320	0	0	9375	31.25%	2.39303
6	Run Mean	0.1007	—	0.0752	1	0	0	0.00%	2.33618
	MSEAT	0.0121	1	0.0203	1	1	16488	54.96%	2.28134
	MSEAET	0.0391	1	0.0056	1	1	21419	71.40%	2.26662
	MSEASVT	0.0161	1	0.0242	1	1	10489	34.96%	2.28516
	MSESET	0.1811	0	0.1182	0	0	10778	35.93%	2.37920
7	Run Mean	0.1660	—	0.1045	1	0	0	0.00%	2.36550
	MSEAT	0.0003	1	0.0263	1	1	22177	73.92%	2.28725
	MSEAET	0.0397	1	0.0304	1	1	23604	78.68%	2.29143
	MSEASVT	0.0072	1	0.0297	1	1	17452	58.17%	2.29070
	MSESET	0.2253	0	0.1972	0	0	10775	35.92%	2.45816
8	Run Mean	0.1315	—	0.0943	1	0	0	0.00%	2.35527
	MSEAT	0.0294	1	0.0426	1	1	23824	79.41%	2.30355
	MSEAET	0.0659	1	0.0492	1	0	24243	80.81%	2.31024
	MSEASVT	0.0308	1	0.0388	1	1	23614	78.71%	2.29979
	MSESET	0.1786	0	0.1528	0	0	8732	29.11%	2.41383
9	Run Mean	0.1185	—	0.0180	1	1	0	0.00%	2.27900
	MSEAT	0.0003	1	0.0264	1	1	21646	72.15%	2.28737
	MSEAET	0.0461	1	0.0087	1	1	22739	75.80%	2.26969
	MSEASVT	0.0234	1	0.0090	1	1	13503	45.01%	2.27000
	MSESET	0.1688	0	-0.0129	1	1	7307	24.36%	2.24806

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.0999	—	0.0479	1	0	0	0.00%	2.30895
	MSEAT	0.0020	1	0.0254	1	1	17995	59.98%	2.28645
	MSEAET	0.0280	1	0.0393	1	1	22276	74.25%	2.30028
	MSEASVT	0.0405	1	0.0046	1	1	13934	46.45%	2.26565
	MSESET	0.1820	0	0.1445	0	0	11042	36.81%	2.40545
11	Run Mean	0.0568	—	0.0716	1	0	0	0.00%	2.33256
	MSEAT	0.0016	1	0.0266	1	1	15829	52.76%	2.28758
	MSEAET	0.0454	1	-0.0114	1	1	21522	71.74%	2.24956
	MSEASVT	0.0318	1	0.0481	1	0	5456	18.19%	2.30909
	MSESET	0.1508	0	0.0220	1	1	11295	37.65%	2.28303
12	Run Mean	0.0777	—	-0.0299	1	1	0	0.00%	2.23110
	MSEAT	0.0000	1	0.0265	1	1	13458	44.86%	2.28748
	MSEAET	0.0127	1	0.0364	1	1	21009	70.03%	2.29741
	MSEASVT	0.0002	1	0.0263	1	1	10883	36.28%	2.28731
	MSESET	0.0773	1	0.0535	1	0	10157	33.86%	2.31447
13	Run Mean	0.1274	—	0.1434	0	0	0	0.00%	2.40443
	MSEAT	0.0026	1	0.0257	1	1	24366	81.22%	2.28667
	MSEAET	0.0317	1	0.0105	1	1	26390	87.97%	2.27155
	MSEASVT	0.0110	1	0.0312	1	1	14975	49.92%	2.29216
	MSESET	0.1495	0	0.1751	0	0	10953	36.51%	2.43613
14	Run Mean	0.1020	—	-0.0478	1	0	0	0.00%	2.21321
	MSEAT	0.0002	1	0.0265	1	1	19002	63.34%	2.28749
	MSEAET	0.0308	1	0.0007	1	1	22031	73.44%	2.26173
	MSEASVT	0.0020	1	0.0250	1	1	15485	51.62%	2.28597
	MSESET	0.1313	0	-0.0776	1	0	8179	27.26%	2.18343
15	Run Mean	0.1487	—	0.1353	0	0	0	0.00%	2.39625
	MSEAT	0.0470	1	0.0028	1	1	20853	69.51%	2.26377
	MSEAET	0.0637	1	0.0012	1	1	24433	81.44%	2.26223
	MSEASVT	0.0488	1	0.0081	1	1	18670	62.23%	2.26907
	MSESET	0.2103	0	0.3062	0	0	9797	32.66%	2.56720
16	Run Mean	0.1816	—	0.1770	0	0	0	0.00%	2.43796
	MSEAT	0.0027	1	0.0243	1	1	22154	73.85%	2.28534
	MSEAET	0.0287	1	0.0075	1	1	24913	83.04%	2.26852
	MSEASVT	0.0071	1	0.0286	1	1	16866	56.22%	2.28956
	MSESET	0.2334	0	0.2102	0	0	8920	29.73%	2.47120
17	Run Mean	0.1030	—	0.0314	1	1	0	0.00%	2.29241
	MSEAT	0.0154	1	0.0189	1	1	17426	58.09%	2.27990
	MSEAET	0.0148	1	0.0261	1	1	22280	74.27%	2.28711
	MSEASVT	0.0161	1	0.0204	1	1	11508	38.36%	2.28141
	MSESET	0.1110	0	0.0826	1	0	6645	22.15%	2.34362
18	Run Mean	0.1509	—	0.1130	1	0	0	0.00%	2.37396
	MSEAT	0.0328	1	0.0010	1	1	15075	50.25%	2.26199
	MSEAET	0.0604	1	-0.0472	1	0	20058	66.86%	2.21381
	MSEASVT	0.0333	1	0.0025	1	1	11165	37.22%	2.26349
	MSESET	0.1339	1	-0.0572	1	0	10111	33.70%	2.20385

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	0.1563	—	0.0736	1	0	0	0.00%	2.33460
	MSEAT	0.0337	1	0.0092	1	1	13864	46.21%	2.27024
	MSEAET	0.0588	1	-0.0013	1	1	18249	60.83%	2.25970
	MSEASVT	0.0343	1	0.0115	1	1	9660	32.20%	2.27247
	MSESET	0.2008	0	-0.0385	1	1	8096	26.99%	2.22248
20	Run Mean	0.1400	—	0.1255	0	0	0	0.00%	2.38652
	MSEAT	0.0007	1	0.0262	1	1	24084	80.28%	2.28721
	MSEAET	0.0270	1	0.0048	1	1	25495	84.98%	2.26579
	MSEASVT	0.0065	1	0.0221	1	1	18368	61.23%	2.28307
	MSESET	0.1929	0	0.1807	0	0	8688	28.96%	2.44167
21	Run Mean	0.0953	—	0.1315	0	0	0	0.00%	2.39252
	MSEAT	0.0002	1	0.0264	1	1	15397	51.32%	2.28738
	MSEAET	0.0309	1	0.0096	1	1	20964	69.88%	2.27058
	MSEASVT	0.0023	1	0.0268	1	1	8058	26.86%	2.28784
	MSESET	0.1720	0	0.2279	0	0	11422	38.07%	2.48891
22	Run Mean	0.0926	—	-0.0348	1	1	0	0.00%	2.22622
	MSEAT	0.0003	1	0.0266	1	1	20384	67.95%	2.28758
	MSEAET	0.0410	1	-0.0126	1	1	23198	77.33%	2.24840
	MSEASVT	0.0126	1	0.0249	1	1	12116	40.39%	2.28593
	MSESET	0.1251	0	-0.0259	1	1	9446	31.49%	2.23515
23	Run Mean	0.0989	—	0.0328	1	1	0	0.00%	2.29376
	MSEAT	0.0237	1	0.0106	1	1	15202	50.67%	2.27157
	MSEAET	0.0520	1	-0.0278	1	1	20882	69.61%	2.23322
	MSEASVT	0.0237	1	0.0106	1	1	11974	39.91%	2.27165
	MSESET	0.1784	0	-0.0409	1	1	10029	33.43%	2.22010
24	Run Mean	0.0690	—	-0.0237	1	1	0	0.00%	2.23730
	MSEAT	0.0128	1	0.0199	1	1	15230	50.77%	2.28094
	MSEAET	0.0610	1	0.0208	1	1	19726	65.75%	2.28179
	MSEASVT	0.0130	1	0.0204	1	1	10115	33.72%	2.28136
	MSESET	0.1573	0	-0.0815	1	0	8605	28.68%	2.17947
25	Run Mean	0.0832	—	0.0249	1	1	0	0.00%	2.28587
	MSEAT	0.0001	1	0.0264	1	1	11644	38.81%	2.28741
	MSEAET	0.0161	1	0.0128	1	1	19697	65.66%	2.27376
	MSEASVT	0.0037	1	0.0243	1	1	10062	33.54%	2.28526
	MSESET	0.1297	0	-0.0538	1	0	9061	30.20%	2.20716

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.1531	—	0.0967	1	0	0	0.00%	2.35768
	MSEAT	0.0054	1	0.0292	1	1	18096	60.32%	2.29020
	MSEAET	0.0197	1	0.0187	1	1	22166	73.89%	2.27968
	MSEASVT	0.0066	1	0.0228	1	1	14641	48.80%	2.28382
	MSESET	0.2301	0	0.0840	1	0	8225	27.42%	2.34500
2	Run Mean	0.1545	—	0.1038	1	0	0	0.00%	2.36476
	MSEAT	0.0004	1	0.0266	1	1	21718	72.39%	2.28764
	MSEAET	0.0424	1	0.0274	1	1	23819	79.40%	2.28839
	MSEASVT	0.0090	1	0.0233	1	1	13716	45.72%	2.28429
	MSESET	0.1384	1	0.0752	1	0	8629	28.76%	2.33619
3	Run Mean	0.1667	—	0.0495	1	0	0	0.00%	2.31047
	MSEAT	0.0243	1	0.0391	1	1	20866	69.55%	2.30011
	MSEAET	0.0648	1	0.0402	1	1	22705	75.68%	2.30122
	MSEASVT	0.0247	1	0.0377	1	1	15462	51.54%	2.29867
	MSESET	0.1681	0	0.0123	1	1	11537	38.46%	2.27326
4	Run Mean	0.0964	—	0.0413	1	1	0	0.00%	2.30231
	MSEAT	0.0157	1	0.0187	1	1	18227	60.76%	2.27974
	MSEAET	0.0405	1	0.0053	1	1	22346	74.49%	2.26631
	MSEASVT	0.0158	1	0.0157	1	1	11212	37.37%	2.27667
	MSESET	0.0811	1	-0.0522	1	0	7515	25.05%	2.20882
5	Run Mean	0.0954	—	0.0460	1	0	0	0.00%	2.30697
	MSEAT	0.0000	1	0.0265	1	1	14330	47.77%	2.28747
	MSEAET	0.0192	1	0.0308	1	1	20103	67.01%	2.29180
	MSEASVT	0.0016	1	0.0256	1	1	10755	35.85%	2.28664
	MSESET	0.1311	0	0.0847	1	0	9159	30.53%	2.34566
6	Run Mean	0.1005	—	0.0889	1	0	0	0.00%	2.34993
	MSEAT	0.0120	1	0.0201	1	1	14870	49.57%	2.28112
	MSEAET	0.0390	1	0.0076	1	1	20644	68.81%	2.26864
	MSEASVT	0.0161	1	0.0248	1	1	10675	35.58%	2.28578
	MSESET	0.1815	0	0.1254	0	0	10736	35.79%	2.38638
7	Run Mean	0.1713	—	0.0955	1	0	0	0.00%	2.35654
	MSEAT	0.0001	1	0.0265	1	1	22830	76.10%	2.28746
	MSEAET	0.0355	1	0.0225	1	1	25274	84.25%	2.28346
	MSEASVT	0.0035	1	0.0242	1	1	18677	62.26%	2.28522
	MSESET	0.2372	0	0.1919	0	0	10580	35.27%	2.45287
8	Run Mean	0.1445	—	0.1088	1	0	0	0.00%	2.36980
	MSEAT	0.0683	1	0.0629	1	0	23719	79.06%	2.32387
	MSEAET	0.0800	1	0.0778	1	0	24024	80.08%	2.33876
	MSEASVT	0.0711	1	0.0543	1	0	22091	73.64%	2.31529
	MSESET	0.1856	0	0.1720	0	0	8212	27.37%	2.43300
9	Run Mean	0.1378	—	0.0412	1	1	0	0.00%	2.30215
	MSEAT	0.0004	1	0.0264	1	1	22220	74.07%	2.28736
	MSEAET	0.0461	1	0.0073	1	1	23469	78.23%	2.26833
	MSEASVT	0.0234	1	0.0093	1	1	14919	49.73%	2.27029
	MSESET	0.1713	0	0.0232	1	1	7224	24.08%	2.28424

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.1049	—	0.0453	1	0	0	0.00%	2.30631
	MSEAT	0.0005	1	0.0262	1	1	19212	64.04%	2.28718
	MSEAET	0.0275	1	0.0412	1	1	22774	75.91%	2.30218
	MSEASVT	0.0023	1	0.0257	1	1	11176	37.25%	2.28674
	MSESET	0.2036	0	0.1557	0	0	10818	36.06%	2.41666
11	Run Mean	0.0617	—	0.0473	1	0	0	0.00%	2.30829
	MSEAT	0.0030	1	0.0250	1	1	16661	55.54%	2.28601
	MSEAET	0.0427	1	-0.0096	1	1	16951	56.50%	2.25139
	MSEASVT	0.0331	1	0.0451	1	1	5317	17.72%	2.30612
	MSESET	0.1612	0	-0.0103	1	1	10932	36.44%	2.25066
12	Run Mean	0.0900	—	0.0072	1	1	0	0.00%	2.26823
	MSEAT	0.0036	1	0.0283	1	1	19565	65.22%	2.28934
	MSEAET	0.0107	1	0.0353	1	1	24116	80.39%	2.29633
	MSEASVT	0.0107	1	0.0383	1	1	9486	31.62%	2.29929
	MSESET	0.1049	0	0.1156	0	0	10871	36.24%	2.37658
13	Run Mean	0.1279	—	0.1261	0	0	0	0.00%	2.38708
	MSEAT	0.0552	1	-0.0013	1	1	18357	61.19%	2.25969
	MSEAET	0.0670	1	-0.0062	1	1	23047	76.82%	2.25475
	MSEASVT	0.0573	1	0.0047	1	1	11556	38.52%	2.26570
	MSESET	0.1578	0	0.1425	0	0	11254	37.51%	2.40351
14	Run Mean	0.0797	—	-0.0128	1	1	0	0.00%	2.24823
	MSEAT	0.0001	1	0.0265	1	1	15787	52.62%	2.28753
	MSEAET	0.0299	1	-0.0024	1	1	21426	71.42%	2.25856
	MSEASVT	0.0006	1	0.0258	1	1	13072	43.57%	2.28680
	MSESET	0.1289	0	-0.0411	1	1	11205	37.35%	2.21990
15	Run Mean	0.1425	—	0.1287	0	0	0	0.00%	2.38970
	MSEAT	0.0244	1	0.0145	1	1	19441	64.80%	2.27549
	MSEAET	0.0461	1	0.0031	1	1	23844	79.48%	2.26408
	MSEASVT	0.0266	1	0.0199	1	1	15260	50.87%	2.28094
	MSESET	0.2047	0	0.3156	0	0	10839	36.13%	2.57661
16	Run Mean	0.1568	—	0.1529	0	0	0	0.00%	2.41393
	MSEAT	0.0007	1	0.0259	1	1	19303	64.34%	2.28692
	MSEAET	0.0289	1	0.0069	1	1	23397	77.99%	2.26793
	MSEASVT	0.0007	1	0.0256	1	1	16840	56.13%	2.28663
	MSESET	0.1628	0	0.1613	0	0	7757	25.86%	2.42233
17	Run Mean	0.1115	—	0.0205	1	1	0	0.00%	2.28152
	MSEAT	0.0039	1	0.0244	1	1	18914	63.05%	2.28544
	MSEAET	0.0154	1	0.0272	1	1	22945	76.48%	2.28821
	MSEASVT	0.0440	1	0.0048	1	1	11127	37.09%	2.26579
	MSESET	0.1616	0	0.0781	1	0	8273	27.58%	2.33906
18	Run Mean	0.1341	—	0.1344	0	0	0	0.00%	2.39540
	MSEAT	0.0225	1	0.0156	1	1	13829	46.10%	2.27665
	MSEAET	0.0490	1	-0.0236	1	1	20021	66.74%	2.23741
	MSEASVT	0.0226	1	0.0148	1	1	10386	34.62%	2.27577
	MSESET	0.1217	1	-0.0239	1	1	9335	31.12%	2.23711

Micro Analysis of Results

Model: U / Ln / 3 Replication Deletion w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	0.1496	—	0.0844	1	0	0	0.00%	2.34540
	MSEAT	0.0337	1	0.0093	1	1	15726	52.42%	2.27035
	MSEAET	0.0604	1	0.0025	1	1	19715	65.72%	2.26348
	MSEASVT	0.0342	1	0.0109	1	1	10126	33.75%	2.27190
	MSESET	0.1992	0	-0.0263	1	1	7198	23.99%	2.23469
20	Run Mean	0.1375	—	0.1138	0	0	0	0.00%	2.37480
	MSEAT	0.0240	1	0.0377	1	1	18279	60.93%	2.29873
	MSEAET	0.0582	1	0.0259	1	1	17199	57.33%	2.28687
	MSEASVT	0.0258	1	0.0333	1	1	15611	52.04%	2.29429
	MSESET	0.1936	0	0.1825	0	0	9751	32.50%	2.44350
21	Run Mean	0.1029	—	0.1457	0	0	0	0.00%	2.40673
	MSEAT	0.0001	1	0.0264	1	1	16177	53.92%	2.28743
	MSEAET	0.0283	1	0.0123	1	1	17210	57.37%	2.27328
	MSEASVT	0.0024	1	0.0264	1	1	10761	35.87%	2.28741
	MSESET	0.1651	0	0.2377	0	0	10468	34.89%	2.49869
22	Run Mean	0.0841	—	-0.0562	1	0	0	0.00%	2.20480
	MSEAT	0.0001	1	0.0265	1	1	15681	52.27%	2.28754
	MSEAET	0.0406	1	-0.0062	1	1	20802	69.34%	2.25475
	MSEASVT	0.0142	1	0.0258	1	1	11457	38.19%	2.28680
	MSESET	0.1012	0	-0.0479	1	0	8741	29.14%	2.21313
23	Run Mean	0.1216	—	0.0708	1	0	0	0.00%	2.33178
	MSEAT	0.0081	1	0.0223	1	1	21646	72.15%	2.28334
	MSEAET	0.0400	1	-0.0162	1	1	24443	81.48%	2.24482
	MSEASVT	0.0081	1	0.0224	1	1	13240	44.13%	2.28337
	MSESET	0.1693	0	0.0175	1	1	10078	33.59%	2.27846
24	Run Mean	0.0615	—	-0.0103	1	1	0	0.00%	2.25071
	MSEAT	0.0128	1	0.0197	1	1	13679	45.60%	2.28068
	MSEAET	0.0636	0	0.0254	1	1	18779	62.60%	2.28642
	MSEASVT	0.0138	1	0.0178	1	1	6537	21.79%	2.27878
	MSESET	0.1564	0	-0.0821	1	0	8520	28.40%	2.17895
25	Run Mean	0.0743	—	0.0356	1	1	0	0.00%	2.29663
	MSEAT	0.0001	1	0.0265	1	1	14789	49.30%	2.28754
	MSEAET	0.0151	1	0.0188	1	1	21738	72.46%	2.27979
	MSEASVT	0.0071	1	0.0213	1	1	7273	24.24%	2.28226
	MSESET	0.1261	0	-0.0420	1	1	8934	29.78%	2.21899

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	14.9801	—	-12.9680	0	0	0	0.00%	79.82801
	MSEAT	4.4954	1	-4.2287	1	1	11200	53.34%	88.56725
	MSEAET	4.5651	1	-4.5230	1	1	11710	55.76%	88.27301
	MSEASVT	4.9144	1	-4.4720	1	1	10491	49.96%	88.32401
	MSESET	22.8613	0	-13.7161	0	0	7136	33.98%	79.07994
2	Run Mean	14.6992	—	-26.9557	0	0	0	0.00%	65.84031
	MSEAT	8.5041	1	-11.3637	0	0	15453	73.59%	81.43226
	MSEAET	8.3696	1	-11.7724	0	0	16128	76.80%	81.02357
	MSEASVT	8.4994	1	-11.3721	0	0	15454	73.59%	81.42388
	MSESET	24.6169	0	-25.3699	0	0	7834	37.30%	67.42605
3	Run Mean	11.3189	—	-22.1734	0	0	0	0.00%	70.62255
	MSEAT	7.4895	1	-8.8201	1	0	12135	57.78%	83.97594
	MSEAET	7.3292	1	-9.5961	0	0	11182	53.25%	83.19986
	MSEASVT	7.4883	1	-8.8225	1	0	11247	53.56%	83.97350
	MSESET	8.5296	1	-33.1382	0	0	8968	42.70%	59.65775
4	Run Mean	6.9661	—	-21.4342	0	0	0	0.00%	71.36177
	MSEAT	7.6629	0	-8.8876	1	0	10346	49.27%	83.90841
	MSEAET	7.3935	0	-9.7850	0	0	10030	47.76%	83.01105
	MSEASVT	7.6612	0	-8.8911	1	0	9317	44.36%	83.90493
	MSESET	17.9035	0	-25.8918	0	0	7298	34.75%	66.90415
5	Run Mean	13.9444	—	-13.1860	0	0	0	0.00%	79.60999
	MSEAT	2.0879	1	-1.1747	1	1	12681	60.39%	91.62126
	MSEAET	2.2970	1	-2.2474	1	1	12115	57.69%	90.54861
	MSEASVT	2.0941	1	-1.4702	1	1	12565	59.83%	91.32579
	MSESET	15.5009	0	-22.9847	0	0	8043	38.30%	69.81132
6	Run Mean	18.9944	—	-4.0270	1	1	0	0.00%	88.76904
	MSEAT	6.8036	1	1.4789	1	1	12358	58.85%	94.27494
	MSEAET	6.9868	1	1.0452	1	1	12774	60.83%	93.84125
	MSEASVT	6.8051	1	1.4636	1	1	11603	55.25%	94.25963
	MSESET	32.1971	0	8.6013	1	0	8346	39.74%	101.39730
7	Run Mean	18.5317	—	-3.6973	1	1	0	0.00%	89.09866
	MSEAT	7.9842	1	-5.9597	1	0	8769	41.76%	86.83633
	MSEAET	8.2914	1	-5.8474	1	0	8713	41.49%	86.94863
	MSEASVT	7.9645	1	-6.0260	1	0	8714	41.49%	86.76995
	MSESET	40.2855	0	-3.6440	1	1	8564	40.78%	89.15201
8	Run Mean	27.8142	—	8.8785	1	0	0	0.00%	101.67448
	MSEAT	0.2843	1	-0.2631	1	1	14724	70.12%	92.53291
	MSEAET	1.1382	1	-0.9094	1	1	15964	76.02%	91.88663
	MSEASVT	0.2841	1	-0.2631	1	1	14724	70.12%	92.53291
	MSESET	24.9722	1	17.9074	0	0	8761	41.72%	110.70342
9	Run Mean	22.6046	—	-11.2363	0	0	0	0.00%	81.55974
	MSEAT	6.6396	1	-5.1933	1	0	14342	68.29%	87.60270
	MSEAET	6.5827	1	-5.6194	1	0	13560	64.57%	87.17657
	MSEASVT	6.6424	1	-5.1846	1	0	14497	69.03%	87.61137
	MSESET	23.0943	0	-8.9765	1	0	8517	40.56%	83.81953

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	14.6792	—	-17.3060	0	0	0	0.00%	75.49003
	MSEAT	4.2286	1	0.1261	1	1	12496	59.50%	92.92206
	MSEAET	4.3784	1	-0.3576	1	1	13113	62.44%	92.43844
	MSEASVT	4.2287	1	0.1221	1	1	11361	54.10%	92.91811
	MSESET	23.3259	0	-9.5298	0	0	9502	45.25%	83.26617
11	Run Mean	25.1743	—	7.6068	1	0	0	0.00%	100.40276
	MSEAT	9.0132	1	3.4488	1	1	13836	65.89%	96.24476
	MSEAET	9.4346	1	2.0804	1	1	13351	63.58%	94.87639
	MSEASVT	9.0109	1	3.4630	1	1	12576	59.89%	96.25904
	MSESET	33.4846	0	16.1674	0	0	7571	36.05%	108.96340
12	Run Mean	25.6295	—	-0.1650	1	1	0	0.00%	92.63105
	MSEAT	3.5871	1	-2.8514	1	1	12943	61.63%	89.94460
	MSEAET	3.5062	1	-3.8101	1	1	12302	58.58%	88.98591
	MSEASVT	3.5863	1	-2.8538	1	1	11668	55.56%	89.94218
	MSESET	27.3362	0	1.8789	1	1	9222	43.91%	94.67488
13	Run Mean	18.5414	—	-9.1790	1	0	0	0.00%	83.61697
	MSEAT	3.8222	1	-0.1653	1	1	15664	74.59%	92.63066
	MSEAET	3.9249	1	-0.4621	1	1	14273	67.97%	92.33389
	MSEASVT	3.8223	1	-0.1738	1	1	15235	72.55%	92.62221
	MSESET	29.1155	0	-5.2470	1	0	9445	44.98%	87.54901
14	Run Mean	8.1413	—	-19.8063	0	0	0	0.00%	72.98974
	MSEAT	4.9803	1	-2.6696	1	1	13580	64.67%	90.12638
	MSEAET	4.9046	1	-3.2838	1	1	12234	58.26%	89.51216
	MSEASVT	4.9760	1	-2.6894	1	1	13254	63.11%	90.10662
	MSESET	10.5293	0	-16.0505	0	0	8331	39.67%	76.74550
15	Run Mean	21.2686	—	-3.6285	1	1	0	0.00%	89.16752
	MSEAT	7.8447	1	-7.0812	1	0	12901	61.44%	85.71484
	MSEAET	7.8645	1	-7.9941	1	0	11339	53.99%	84.80188
	MSEASVT	7.8444	1	-7.0819	1	0	11880	56.57%	85.71413
	MSESET	26.9649	0	-4.5243	1	1	9074	43.21%	88.27171
16	Run Mean	26.3079	—	7.3637	1	0	0	0.00%	100.15969
	MSEAT	5.1132	1	-1.6188	1	1	9745	46.40%	91.17723
	MSEAET	5.3327	1	-1.8052	1	1	10941	52.10%	90.99082
	MSEASVT	5.1116	1	-1.6322	1	1	8770	41.76%	91.16379
	MSESET	51.5978	0	35.9489	0	0	8203	39.06%	128.74486
17	Run Mean	16.7730	—	4.9164	1	0	0	0.00%	97.71242
	MSEAT	0.5638	1	-0.1172	1	1	12871	61.29%	92.67878
	MSEAET	1.7421	1	-0.0155	1	1	12508	59.56%	92.78054
	MSEASVT	0.5639	1	-0.1248	1	1	12725	60.60%	92.67125
	MSESET	25.8116	0	10.4263	0	0	6477	30.84%	103.22230
18	Run Mean	23.4182	—	-11.2791	0	0	0	0.00%	81.51693
	MSEAT	10.1653	1	-7.3945	1	0	12391	59.00%	85.40152
	MSEAET	10.0885	1	-7.9454	1	0	12830	61.10%	84.85059
	MSEASVT	10.1596	1	-7.4134	1	0	11373	54.15%	85.38259
	MSESET	32.7284	0	2.8446	1	1	8330	39.66%	95.64056

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	22.7128	—	8.5976	1	0	0	0.00%	101.39363
	MSEAT	4.3774	1	-2.5621	1	1	13007	61.94%	90.23386
	MSEAET	4.7316	1	-2.6780	1	1	12031	57.29%	90.11796
	MSEASVT	4.3731	1	-2.5946	1	1	12950	61.67%	90.20136
	MSESET	34.7549	0	16.4683	0	0	7608	36.23%	109.26432
20	Run Mean	20.6435	—	-8.2843	1	0	0	0.00%	84.51167
	MSEAT	8.9824	1	-9.1772	1	0	13418	63.89%	83.61881
	MSEAET	8.7126	1	-9.9859	0	0	12239	58.28%	82.81008
	MSEASVT	8.9615	1	-9.2324	1	0	12019	57.23%	83.56355
	MSESET	38.5857	0	-9.5048	0	0	8457	40.27%	83.29119
21	Run Mean	16.8303	—	-7.6721	1	0	0	0.00%	85.12390
	MSEAT	6.0708	1	-5.8314	1	0	10491	49.96%	86.96460
	MSEAET	6.0037	1	-6.2806	1	0	10340	49.24%	86.51541
	MSEASVT	6.0693	1	-5.8350	1	0	10351	49.29%	86.96099
	MSESET	11.5258	1	-20.8346	0	0	7756	36.93%	71.96142
22	Run Mean	32.1562	—	7.1478	1	0	0	0.00%	99.94377
	MSEAT	5.0362	1	-1.9777	1	1	14362	68.39%	90.81832
	MSEAET	5.2544	1	-2.8818	1	1	13958	66.47%	89.91423
	MSEASVT	5.0313	1	-2.0242	1	1	14360	68.38%	90.77184
	MSESET	53.6521	0	12.4131	0	0	8314	39.59%	105.20914
23	Run Mean	27.6855	—	4.7366	1	0	0	0.00%	97.53264
	MSEAT	7.1606	1	-7.7499	1	0	12930	61.57%	85.04606
	MSEAET	7.2339	1	-7.8161	1	0	13177	62.75%	84.97990
	MSEASVT	7.1613	1	-7.7499	1	0	12830	61.10%	85.04610
	MSESET	46.6272	0	21.7023	0	0	8384	39.92%	114.49832
24	Run Mean	16.7995	—	-7.4308	1	0	0	0.00%	85.36515
	MSEAT	8.2099	1	-0.9649	1	1	8577	40.84%	91.83111
	MSEAET	8.2861	1	-1.6720	1	1	8925	42.50%	91.12404
	MSEASVT	8.2096	1	-0.9714	1	1	9048	43.08%	91.82455
	MSESET	22.1770	0	-12.3668	0	0	8829	42.04%	80.42918
25	Run Mean	31.2071	—	32.1117	0	0	0	0.00%	124.90774
	MSEAT	5.1823	1	1.1536	1	1	14387	68.51%	93.94958
	MSEAET	5.3588	1	1.2229	1	1	11775	56.07%	94.01885
	MSEASVT	5.1823	1	1.1541	1	1	13730	65.38%	93.95006

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
1	Run Mean	14.2455	—	-16.2175	0	0	0	0.00%	76.5785
	MSEAT	4.7847	1	-3.7765	1	1	14143	67.35%	89.0195
	MSEAET	4.7240	1	-4.1524	1	1	13579	64.66%	88.6436
	MSEASVT	4.8040	1	-3.7199	1	1	12517	59.61%	89.0761
	MSESET	22.0084	0	-14.7008	0	0	7389	35.19%	78.0952
2	Run Mean	12.9153	—	-29.9097	0	0	0	0.00%	62.8863
	MSEAT	8.2623	1	-11.5430	0	0	17460	83.14%	81.2530
	MSEAET	8.2314	1	-11.7213	0	0	16691	79.48%	81.0747
	MSEASVT	8.2565	1	-11.5528	0	0	17460	83.14%	81.2432
	MSESET	24.2238	0	-25.4769	0	0	8266	39.36%	67.3191
3	Run Mean	11.3392	—	-21.7849	0	0	0	0.00%	71.0111
	MSEAT	7.0771	1	-7.8551	1	0	11523	54.87%	84.9409
	MSEAET	6.9345	1	-8.4769	1	0	10678	50.85%	84.3191
	MSEASVT	7.0737	1	-7.8625	1	0	10874	51.78%	84.9335
	MSESET	9.3859	1	-32.5682	0	0	9468	45.09%	60.2278
4	Run Mean	7.4771	—	-25.1476	0	0	0	0.00%	67.6484
	MSEAT	9.3036	0	-11.4861	0	0	12579	59.90%	81.3099
	MSEAET	9.0451	0	-12.3049	0	0	12175	57.98%	80.4911
	MSEASVT	9.2967	0	-11.4995	0	0	11240	53.52%	81.2965
	MSESET	18.3246	0	-26.1649	0	0	6218	29.61%	66.6311
5	Run Mean	14.5080	—	-13.9343	0	0	0	0.00%	78.8617
	MSEAT	2.2689	1	-1.4794	1	1	12701	60.48%	91.3166
	MSEAET	2.5229	1	-2.6637	1	1	11821	56.29%	90.1323
	MSEASVT	2.2654	1	-1.4933	1	1	12533	59.68%	91.3027
	MSESET	17.0950	0	-20.9896	0	0	7551	35.96%	71.8064
6	Run Mean	17.8837	—	-9.1564	1	0	0	0.00%	83.6396
	MSEAT	6.8562	1	2.0140	1	1	13450	64.05%	94.8100
	MSEAET	7.1233	1	1.4584	1	1	12996	61.89%	94.2544
	MSEASVT	6.8578	1	2.0014	1	1	12625	60.12%	94.7974
	MSESET	32.0449	0	8.4911	1	0	8378	39.89%	101.2871
7	Run Mean	20.7924	—	-11.5000	0	0	0	0.00%	81.2960
	MSEAT	9.6511	1	-8.5359	1	0	9449	45.00%	84.2601
	MSEAET	9.8493	1	-8.2508	1	0	9979	47.52%	84.5452
	MSEASVT	9.4864	1	-8.3311	1	0	11503	54.78%	84.4649
	MSESET	39.9157	0	-3.7913	1	1	8598	40.94%	89.0047
8	Run Mean	26.1807	—	7.1613	1	0	0	0.00%	99.9573
	MSEAT	0.0078	1	-0.1256	1	1	16204	77.16%	92.6704
	MSEAET	0.7783	1	-0.6852	1	1	16219	77.23%	92.1108
	MSEASVT	0.1941	1	-0.0210	1	1	12525	59.64%	92.7750
	MSESET	25.2913	1	18.8437	0	0	8690	41.38%	111.6397
9	Run Mean	22.0318	—	-12.6623	0	0	0	0.00%	80.1337
	MSEAT	6.6283	1	-4.9466	1	0	15383	73.25%	87.8494
	MSEAET	6.5797	1	-5.1323	1	0	14371	68.44%	87.6637
	MSEASVT	6.6230	1	-4.9638	1	0	14469	68.90%	87.8322
	MSESET	22.5280	0	-10.2739	0	0	8720	41.53%	82.5221

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
10	Run Mean	14.5300	—	-17.9665	0	0	0	0.00%	74.8295
	MSEAT	7.7437	1	-1.7904	1	1	15177	72.27%	91.0056
	MSEAET	7.7345	1	-1.9464	1	1	14340	68.28%	90.8496
	MSEASVT	7.7339	1	-1.9682	1	1	11445	54.50%	90.8278
	MSESET	23.1093	0	-10.0503	0	0	9578	45.61%	82.7457
11	Run Mean	25.0200	—	5.7627	1	0	0	0.00%	98.5587
	MSEAT	9.5252	1	4.0272	1	1	11870	56.52%	96.8232
	MSEAET	9.9384	1	2.7180	1	1	12688	60.42%	95.5140
	MSEASVT	9.5263	1	4.0211	1	1	11866	56.51%	96.8171
	MSESET	33.7161	0	16.8018	0	0	7507	35.75%	109.5978
12	Run Mean	26.1262	—	-2.4087	1	1	0	0.00%	90.3873
	MSEAT	3.9909	1	-4.6911	1	0	11920	56.76%	88.1049
	MSEAET	4.1162	1	-5.2140	1	0	11774	56.07%	87.5820
	MSEASVT	3.9908	1	-4.6913	1	0	11171	53.19%	88.1047
	MSESET	27.7895	0	0.2480	1	1	9091	43.29%	93.0440
13	Run Mean	14.7794	—	-15.0389	0	0	0	0.00%	77.7571
	MSEAT	2.5565	1	-0.7964	1	1	15224	72.49%	91.9996
	MSEAET	3.0496	1	-0.9695	1	1	14708	70.04%	91.8265
	MSEASVT	2.5496	1	-0.9110	1	1	14871	70.81%	91.8850
	MSESET	27.5567	0	-5.6343	1	0	8393	39.97%	87.1617
14	Run Mean	9.2595	—	-21.0262	0	0	0	0.00%	71.7698
	MSEAT	5.6124	1	-2.9872	1	1	12490	59.47%	89.8088
	MSEAET	5.5369	1	-3.5579	1	1	11401	54.29%	89.2381
	MSEASVT	5.6057	1	-3.0184	1	1	12489	59.47%	89.7776
	MSESET	10.9760	0	-13.5521	0	0	9062	43.15%	79.2439
15	Run Mean	20.5139	—	-5.8659	1	0	0	0.00%	86.9301
	MSEAT	8.1553	1	-7.4548	1	0	12615	60.07%	85.3412
	MSEAET	8.1012	1	-8.2743	1	0	11579	55.14%	84.5217
	MSEASVT	8.1536	1	-7.4591	1	0	11922	56.77%	85.3369
	MSESET	27.3870	0	-4.3560	1	1	9378	44.66%	88.4400
16	Run Mean	23.3923	—	4.2525	1	1	0	0.00%	97.0485
	MSEAT	5.0982	1	-1.3316	1	1	11353	54.06%	91.4644
	MSEAET	5.2467	1	-1.5696	1	1	11508	54.80%	91.2264
	MSEASVT	5.0979	1	-1.3349	1	1	8555	40.74%	91.4611
	MSESET	50.2919	0	35.8103	0	0	8177	38.94%	128.6063
17	Run Mean	16.7575	—	-0.9300	1	1	0	0.00%	91.8660
	MSEAT	3.6797	1	-1.9958	1	1	12766	60.79%	90.8002
	MSEAET	3.7989	1	-1.6841	1	1	12739	60.66%	91.1119
	MSEASVT	3.7198	1	-1.8595	1	1	12658	60.28%	90.9365
	MSESET	23.9246	0	9.8371	0	0	6773	32.25%	102.6331
18	Run Mean	22.9417	—	-11.9697	0	0	0	0.00%	80.8263
	MSEAT	9.8392	1	-7.4232	1	0	12055	57.40%	85.3728
	MSEAET	9.7955	1	-7.8278	1	0	12417	59.13%	84.9682
	MSEASVT	9.8008	1	-7.4083	1	0	10777	51.32%	85.3877
	MSESET	32.6527	0	3.6586	1	1	9104	43.35%	96.4546

Micro Analysis of Results

Model: E / E / 4 Replication Deletion w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Avg Trunc Pt.	% Trunc	Estimate
19	Run Mean	22.1293	—	4.7941	1	0	0	0.00%	97.5901
	MSEAT	5.9075	1	-0.0022	1	1	14302	68.10%	92.7938
	MSEAET	6.1027	1	-0.7107	1	1	13540	64.47%	92.0853
	MSEASVT	5.9078	1	0.0401	1	1	14301	68.10%	92.8361
	MSESET	35.6543	0	18.5180	0	0	7636	36.36%	111.3140
20	Run Mean	19.7883	—	-10.9865	0	0	0	0.00%	81.8095
	MSEAT	9.3213	1	-9.8104	0	0	13481	64.20%	82.9856
	MSEAET	9.1934	1	-10.2780	0	0	11749	55.95%	82.5180
	MSEASVT	9.3236	1	-9.8053	0	0	11871	56.53%	82.9907
	MSESET	38.1740	0	-10.5651	0	0	8652	41.20%	82.2309
21	Run Mean	16.3289	—	-8.8504	1	0	0	0.00%	83.9456
	MSEAT	5.0246	1	-4.7160	1	0	14704	70.02%	88.0800
	MSEAET	5.0303	1	-5.1205	1	0	13269	63.19%	87.6755
	MSEASVT	5.0209	1	-4.7256	1	0	12845	61.17%	88.0704
	MSESET	11.7685	1	-19.9066	0	0	9145	43.55%	72.8894
22	Run Mean	32.3855	—	4.8887	1	0	0	0.00%	97.6847
	MSEAT	6.1324	1	-5.3359	1	0	12881	61.34%	87.4601
	MSEAET	6.2714	1	-5.7648	1	0	12871	61.29%	87.0312
	MSEASVT	6.1343	1	-5.3306	1	0	12881	61.34%	87.4654
	MSESET	53.4926	0	13.5338	0	0	8320	39.62%	106.3298
23	Run Mean	26.8490	—	3.7195	1	1	0	0.00%	96.5155
	MSEAT	7.3578	1	-8.5278	1	0	13006	61.94%	84.2682
	MSEAET	7.4556	1	-8.3355	1	0	13358	63.61%	84.4605
	MSEASVT	7.3562	1	-8.5312	1	0	12890	61.38%	84.2648
	MSESET	44.8060	0	20.6915	0	0	8466	40.32%	113.4875
24	Run Mean	15.5075	—	-11.2410	0	0	0	0.00%	81.5550
	MSEAT	7.4377	1	-0.6782	1	1	9406	44.79%	92.1178
	MSEAET	7.5268	1	-1.3338	1	1	8986	42.79%	91.4622
	MSEASVT	7.4376	1	-0.6801	1	1	8344	39.74%	92.1159
	MSESET	22.3444	0	-12.5685	0	0	8210	39.10%	80.2275
25	Run Mean	29.3513	—	24.0536	0	0	0	0.00%	116.8496
	MSEAT	5.9305	1	0.6171	1	1	14919	71.04%	93.4131
	MSEAET	6.1991	1	0.8611	1	1	13007	61.94%	93.6571
	MSEASVT	5.9309	1	0.6098	1	1	14756	70.27%	93.4058
	MSESET	32.3733	0	30.8444	0	0	5728	27.27%	123.6404

Micro Analysis of Results

Model: E_2 / E_2 / 4 Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.9139	—	-0.2979	1	0	0	0.00%	13.14209
	MSEAT	3.8195	0	-0.0328	1	1	202343	96.35%	13.40715
	MSEAET	2.0692	0	-0.0486	1	1	170148	81.02%	13.39136
	MSEASVT	0.8436	1	-0.1151	1	1	11105	5.29%	13.32494
	MSESET	0.7850	1	-0.8278	0	0	120834	57.54%	12.61223
2	Run Mean	0.8841	—	-0.2337	1	1	0	0.00%	13.20626
	MSEAT	1.3269	0	-0.0571	1	1	74987	35.71%	13.38285
	MSEAET	0.8867	0	-0.0572	1	1	131923	62.82%	13.38283
	MSEASVT	1.2752	0	-0.0573	1	1	74391	35.42%	13.38268
	MSESET	0.9719	0	0.3185	1	0	92984	44.28%	13.75853
3	Run Mean	0.7900	—	-0.2978	1	0	0	0.00%	13.14223
	MSEAT	2.7412	0	-0.0572	1	1	175288	83.47%	13.38279
	MSEAET	2.4067	0	-0.0611	1	1	182729	87.01%	13.37886
	MSEASVT	1.1190	0	-0.0623	1	1	60869	28.99%	13.37765
	MSESET	0.7034	1	-0.4950	1	0	53738	25.59%	12.94498
4	Run Mean	1.0734	—	0.8799	0	0	0	0.00%	14.31989
	MSEAT	2.1898	0	-0.0570	1	1	166336	79.21%	13.38296
	MSEAET	1.3095	0	-0.0573	1	1	158676	75.56%	13.38266
	MSEASVT	1.2418	0	-0.0472	1	1	123137	58.64%	13.39281
	MSESET	1.1548	0	0.6882	0	0	105817	50.39%	14.12816
5	Run Mean	1.1053	—	-0.3667	1	0	0	0.00%	13.07331
	MSEAT	2.5412	0	-0.0578	1	1	143443	68.31%	13.38217
	MSEAET	2.1872	0	0.1910	1	1	130694	62.24%	13.63103
	MSEASVT	2.5357	0	-0.0628	1	1	139688	66.52%	13.37717
	MSESET	0.9489	1	-0.6348	1	0	29175	13.89%	12.80515
6	Run Mean	0.9751	—	0.5862	1	0	0	0.00%	14.02621
	MSEAT	1.0408	0	-0.0573	1	1	53924	25.68%	13.38269
	MSEAET	1.0584	0	-0.0712	1	1	90744	43.21%	13.36879
	MSEASVT	0.9121	1	-0.0573	1	1	52636	25.06%	13.38267
	MSESET	0.9718	1	0.1390	1	1	40777	19.42%	13.57898
7	Run Mean	1.1204	—	0.0402	1	1	0	0.00%	13.48018
	MSEAT	1.0520	1	-0.0571	1	1	74595	35.52%	13.38285
	MSEAET	1.2509	0	-0.0818	1	1	147647	70.31%	13.35824
	MSEASVT	0.9201	1	-0.0573	1	1	67712	32.24%	13.38272
	MSESET	0.8304	1	0.0402	1	1	23779	11.32%	13.48019
8	Run Mean	1.1891	—	0.2312	1	1	0	0.00%	13.67115
	MSEAT	1.7998	0	-0.0568	1	1	112491	53.57%	13.38323
	MSEAET	1.1436	1	-0.0569	1	1	118628	56.49%	13.38313
	MSEASVT	1.6237	0	-0.0573	1	1	109890	52.33%	13.38266
	MSESET	0.9619	1	0.0191	1	1	79724	37.96%	13.45915
9	Run Mean	0.8816	—	-0.6317	1	0	0	0.00%	12.80827
	MSEAT	3.6914	0	-0.0633	1	1	202043	96.21%	13.37673
	MSEAET	1.7833	0	-0.0603	1	1	161807	77.05%	13.37974
	MSEASVT	1.1158	0	-0.2187	1	1	120156	57.22%	13.22126
	MSESET	0.6111	1	-0.5429	1	0	11429	5.44%	12.89710

Micro Analysis of Results

Model: E_2 / E_2 / 4 Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.7079	—	-0.2023	1	1	0	0.00%	13.23774
	MSEAT	1.7281	0	-0.0579	1	1	164001	78.10%	13.38211
	MSEAET	1.6200	0	-0.0575	1	1	185993	88.57%	13.38248
	MSEASVT	0.9101	0	-0.0752	1	1	24033	11.44%	13.36481
	MSESET	0.5943	1	0.3369	1	0	106169	50.56%	13.77687
11	Run Mean	0.9947	—	0.3019	1	0	0	0.00%	13.74194
	MSEAT	1.1662	0	-0.0572	1	1	94382	44.94%	13.38276
	MSEAET	1.4610	0	0.0602	1	1	156831	74.68%	13.50019
	MSEASVT	0.9968	0	-0.0572	1	1	33605	16.00%	13.38276
	MSESET	0.7684	1	-0.1020	1	1	41127	19.58%	13.33803
12	Run Mean	0.8670	—	-0.4612	1	0	0	0.00%	12.97880
	MSEAT	3.6777	0	-0.0794	1	1	203123	96.73%	13.36061
	MSEAET	1.7855	0	-0.0709	1	1	149499	71.19%	13.36914
	MSEASVT	2.6353	0	-0.0638	1	1	196159	93.41%	13.37617
	MSESET	0.6609	1	-0.9251	0	0	41552	19.79%	12.51494
13	Run Mean	0.8498	—	0.3641	1	0	0	0.00%	13.80407
	MSEAT	1.3186	0	-0.0581	1	1	135825	64.68%	13.38192
	MSEAET	0.7668	1	-0.0580	1	1	129851	61.83%	13.38205
	MSEASVT	1.1548	0	-0.0580	1	1	125596	59.81%	13.38203
	MSESET	0.6589	1	0.4007	1	0	17569	8.37%	13.84067
14	Run Mean	1.0424	—	0.9249	0	0	0	0.00%	14.36489
	MSEAT	1.6086	0	-0.0571	1	1	138619	66.01%	13.38293
	MSEAET	1.7217	0	-0.0593	1	1	173841	82.78%	13.38073
	MSEASVT	1.1689	0	-0.0263	1	1	109948	52.36%	13.41367
	MSESET	0.8507	1	0.2255	1	1	95470	45.46%	13.66547
15	Run Mean	0.7844	—	-0.0063	1	1	0	0.00%	13.43372
	MSEAT	2.4820	0	-0.0511	1	1	6770	3.22%	13.38893
	MSEAET	1.9487	0	0.6603	1	0	105601	50.29%	14.10032
	MSEASVT	0.7639	1	-0.0572	1	1	6234	2.97%	13.38284
	MSESET	0.9328	0	0.5508	1	0	62768	29.89%	13.99076
16	Run Mean	1.0681	—	0.0639	1	1	0	0.00%	13.50387
	MSEAT	2.8207	0	-0.0601	1	1	189683	90.33%	13.37988
	MSEAET	2.0489	0	-0.0616	1	1	191476	91.18%	13.37842
	MSEASVT	1.0338	1	-0.0575	1	1	5679	2.70%	13.38253
	MSESET	0.7985	1	-0.2480	1	1	14731	7.01%	13.19203
17	Run Mean	0.9866	—	0.5412	1	0	0	0.00%	13.98124
	MSEAT	3.5217	0	-0.0103	1	1	198166	94.36%	13.42970
	MSEAET	2.0869	0	0.1938	1	1	153876	73.27%	13.63384
	MSEASVT	3.2581	0	-0.0225	1	1	197114	93.86%	13.41747
	MSESET	1.2531	0	0.9906	0	0	114229	54.39%	14.43061
18	Run Mean	1.1426	—	-0.2834	1	0	0	0.00%	13.15660
	MSEAT	1.7199	0	-0.0582	1	1	156639	74.59%	13.38177
	MSEAET	0.8394	1	-0.0821	1	1	136906	65.19%	13.35791
	MSEASVT	1.4516	0	-0.0578	1	1	22020	10.49%	13.38221
	MSESET	0.9971	1	-1.0461	0	0	96648	46.02%	12.39389

Micro Analysis of Results

Model: $E_2 / E_2 / 4$ Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
19	Run Mean	1.1870	—	0.9963	0	0	0	0.00%	14.43628
	MSEAT	1.8286	0	-0.0573	1	1	148469	70.70%	13.38273
	MSEAET	1.1066	1	-0.0578	1	1	135130	64.35%	13.38216
	MSEASVT	1.8285	0	-0.0570	1	1	148468	70.70%	13.38297
	MSESET	0.9368	1	0.9480	0	0	12637	6.02%	14.38803
20	Run Mean	0.9467	—	0.4559	1	0	0	0.00%	13.89590
	MSEAT	1.9763	0	-0.0572	1	1	149313	71.10%	13.38282
	MSEAET	0.8414	1	-0.0556	1	1	142228	67.73%	13.38441
	MSEASVT	1.7972	0	-0.0571	1	1	123737	58.92%	13.38290
	MSESET	0.8598	1	0.6847	0	0	59314	28.24%	14.12466
21	Run Mean	1.0275	—	-0.7257	0	0	0	0.00%	12.71427
	MSEAT	1.9322	0	-0.0584	1	1	172597	82.19%	13.38161
	MSEAET	1.0167	1	-0.0583	1	1	117111	55.77%	13.38167
	MSEASVT	1.9220	0	-0.0580	1	1	156482	74.52%	13.38202
	MSESET	0.8653	1	-0.3811	1	0	77310	36.81%	13.05890
22	Run Mean	1.1046	—	0.5037	1	0	0	0.00%	13.94368
	MSEAT	1.7130	0	-0.0543	1	1	175312	83.48%	13.38570
	MSEAET	1.4093	0	0.4475	1	0	157208	74.86%	13.88746
	MSEASVT	1.7986	0	-0.0571	1	1	160567	76.46%	13.38287
	MSESET	1.3188	0	1.2107	0	0	104522	49.77%	14.65067
23	Run Mean	0.9750	—	-0.1257	1	1	0	0.00%	13.31432
	MSEAT	0.9845	0	-0.0572	1	1	8103	3.86%	13.38278
	MSEAET	1.2756	0	-0.0520	1	1	131974	62.84%	13.38798
	MSEASVT	0.9310	1	-0.0574	1	1	4651	2.21%	13.38261
	MSESET	0.6308	1	-0.1518	1	1	45104	21.48%	13.28819
24	Run Mean	0.7874	—	-0.2801	1	0	0	0.00%	13.15985
	MSEAT	1.0178	0	-0.0575	1	1	49738	23.68%	13.38249
	MSEAET	1.5851	0	-0.0585	1	1	148018	70.48%	13.38147
	MSEASVT	0.9023	0	-0.0634	1	1	25662	12.22%	13.37660
	MSESET	0.7976	0	0.4094	1	0	103124	49.11%	13.84938
25	Run Mean	1.1146	—	0.6965	0	0	0	0.00%	14.13650
	MSEAT	6.4545	0	-0.0523	1	1	133205	63.43%	13.38767
	MSEAET	2.0447	0	-0.1032	1	1	115494	55.00%	13.33685
	MSEASVT	1.0476	1	0.6713	1	0	120812	57.53%	14.11133
	MSESET	0.7997	1	1.2181	0	0	15739	7.49%	14.65806

Micro Analysis of Results

Model: E_2 / E_2 / 4 Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.9202	—	-0.2983	1	0	0	0.00%	13.14168
	MSEAT	3.7956	0	-0.0561	1	1	202353	96.36%	13.38393
	MSEAET	2.0156	0	-0.0511	1	1	170396	81.14%	13.38891
	MSEASVT	0.8436	1	-0.1153	1	1	11105	5.29%	13.32468
	MSESET	0.7838	1	-0.8272	0	0	120833	57.54%	12.61277
2	Run Mean	0.9318	—	-0.0041	1	1	0	0.00%	13.43585
	MSEAT	0.8455	1	-0.0574	1	1	20999	10.00%	13.38264
	MSEAET	1.0907	0	-0.0560	1	1	116831	55.63%	13.38399
	MSEASVT	0.8511	1	-0.0574	1	1	20220	9.63%	13.38264
	MSESET	0.5874	1	0.1582	1	1	44345	21.12%	13.59819
3	Run Mean	0.7857	—	-0.5719	1	0	0	0.00%	12.86810
	MSEAT	3.5826	0	-0.0574	1	1	180394	85.90%	13.38262
	MSEAET	1.7999	0	-0.1253	1	1	149644	71.26%	13.31470
	MSEASVT	2.6013	0	-0.0765	1	1	164127	78.16%	13.36351
	MSESET	1.1176	0	-1.0198	0	0	97542	46.45%	12.42021
4	Run Mean	1.0581	—	0.9685	0	0	0	0.00%	14.40845
	MSEAT	2.6628	0	-0.0577	1	1	163923	78.06%	13.38232
	MSEAET	1.5493	0	-0.0497	1	1	180975	86.18%	13.39029
	MSEASVT	2.5780	0	-0.0571	1	1	162233	77.25%	13.38285
	MSESET	1.2225	0	0.8611	0	0	68320	32.53%	14.30107
5	Run Mean	0.9510	—	-0.5212	1	0	0	0.00%	12.91879
	MSEAT	4.1461	0	-0.0597	1	1	190461	90.70%	13.38029
	MSEAET	2.4156	0	-0.0580	1	1	159244	75.83%	13.38196
	MSEASVT	2.1944	0	-0.1675	1	1	147642	70.31%	13.27253
	MSESET	0.6656	1	-0.9850	0	0	72938	34.73%	12.45498
6	Run Mean	0.9751	—	0.5862	1	0	0	0.00%	14.02621
	MSEAT	1.0408	0	-0.0573	1	1	53924	25.68%	13.38269
	MSEAET	1.0584	0	-0.0712	1	1	90744	43.21%	13.36879
	MSEASVT	0.9121	1	-0.0573	1	1	52636	25.06%	13.38267
	MSESET	0.9718	1	0.1390	1	1	40777	19.42%	13.57898
7	Run Mean	1.0744	—	0.1984	1	1	0	0.00%	13.63844
	MSEAT	4.8389	0	-0.3562	1	0	209096	99.57%	13.08375
	MSEAET	1.8496	0	-0.1570	1	1	144269	68.70%	13.28303
	MSEASVT	1.3620	0	-0.0576	1	1	111277	52.99%	13.38236
	MSESET	0.7943	1	0.3309	1	0	6675	3.18%	13.77091
8	Run Mean	1.2224	—	0.7701	0	0	0	0.00%	14.21006
	MSEAT	3.0290	0	-0.0570	1	1	201344	95.88%	13.38301
	MSEAET	1.9120	0	-0.0548	1	1	144771	68.94%	13.38517
	MSEASVT	1.5535	0	-0.0572	1	1	110199	52.48%	13.38281
	MSESET	1.2351	0	0.1328	1	1	106234	50.59%	13.57284
9	Run Mean	0.8816	—	-0.6317	1	0	0	0.00%	12.80827
	MSEAT	3.6914	0	-0.0633	1	1	202043	96.21%	13.37673
	MSEAET	1.7833	0	-0.0603	1	1	161807	77.05%	13.37974
	MSEASVT	1.1158	0	-0.2187	1	1	120156	57.22%	13.22126
	MSESET	0.6111	1	-0.5429	1	0	11429	5.44%	12.89710

Micro Analysis of Results

Model: E_2 / E_2 / 4 Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.8223	—	-0.2150	1	1	0	0.00%	13.22502
	MSEAT	1.7571	0	-0.0579	1	1	132741	63.21%	13.38208
	MSEAET	1.4483	0	-0.0693	1	1	165722	78.92%	13.37066
	MSEASVT	0.8757	0	-0.0573	1	1	20386	9.71%	13.38265
	MSESET	0.8673	0	0.3135	1	0	95277	45.37%	13.75353
11	Run Mean	1.0633	—	0.8278	0	0	0	0.00%	14.26784
	MSEAT	2.2663	0	-0.0498	1	1	203534	96.92%	13.39016
	MSEAET	1.2566	0	-0.0547	1	1	132034	62.87%	13.38534
	MSEASVT	1.9517	0	-0.0579	1	1	174795	83.24%	13.38208
	MSESET	0.6904	1	0.9740	0	0	55464	26.41%	14.41402
12	Run Mean	0.8670	—	-0.4612	1	0	0	0.00%	12.97880
	MSEAT	3.6777	0	-0.0794	1	1	203123	96.73%	13.36061
	MSEAET	1.7855	0	-0.0709	1	1	149499	71.19%	13.36914
	MSEASVT	2.6353	0	-0.0638	1	1	196159	93.41%	13.37617
	MSESET	0.6609	1	-0.9251	0	0	41552	19.79%	12.51494
13	Run Mean	0.8539	—	0.3633	1	0	0	0.00%	13.80330
	MSEAT	1.1485	0	-0.0577	1	1	136933	65.21%	13.38226
	MSEAET	0.7740	1	-0.0585	1	1	130614	62.20%	13.38150
	MSEASVT	1.1550	0	-0.0587	1	1	125596	59.81%	13.38129
	MSESET	0.6585	1	0.4002	1	0	17569	8.37%	13.84018
14	Run Mean	0.9541	—	0.5062	1	0	0	0.00%	13.94623
	MSEAT	2.7396	0	-0.0575	1	1	180759	86.08%	13.38250
	MSEAET	1.3673	0	-0.0573	1	1	135006	64.29%	13.38265
	MSEASVT	1.2294	0	-0.0570	1	1	85102	40.52%	13.38299
	MSESET	0.9471	1	0.0585	1	1	69470	33.08%	13.49845
15	Run Mean	0.7844	—	-0.0063	1	1	0	0.00%	13.43372
	MSEAT	2.4820	0	-0.0511	1	1	175843	83.73%	13.38893
	MSEAET	1.9487	0	0.6603	1	0	164531	78.35%	14.10032
	MSEASVT	0.7639	1	-0.0572	1	1	3295	1.57%	13.38284
	MSESET	0.9328	0	0.5508	1	0	97897	46.62%	13.99076
16	Run Mean	1.0682	—	0.0638	1	1	0	0.00%	13.50382
	MSEAT	2.7973	0	-0.0605	1	1	189567	90.27%	13.37948
	MSEAET	2.0014	0	-0.0619	1	1	191438	91.16%	13.37815
	MSEASVT	1.0338	1	-0.0575	1	1	5679	2.70%	13.38248
	MSESET	0.7985	1	-0.2480	1	1	14731	7.01%	13.19197
17	Run Mean	1.0582	—	0.9048	0	0	0	0.00%	14.34478
	MSEAT	4.2827	0	-0.0581	1	1	199842	95.16%	13.38194
	MSEAET	2.3569	0	-0.0572	1	1	163681	77.94%	13.38277
	MSEASVT	2.7588	0	-0.0536	1	1	189987	90.47%	13.38637
	MSESET	1.0665	0	1.2532	0	0	49010	23.34%	14.69322
18	Run Mean	0.9764	—	-0.1438	1	1	0	0.00%	13.29616
	MSEAT	1.0005	0	-0.0573	1	1	12024	5.73%	13.38268
	MSEAET	2.0603	0	-0.0650	1	1	106091	50.52%	13.37496
	MSEASVT	0.9789	0	-0.0574	1	1	10935	5.21%	13.38262
	MSESET	0.9263	1	-0.8469	0	0	97115	46.25%	12.59310

Micro Analysis of Results

Model: $E_2 / E_2 / 4$ Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
19	Run Mean	1.1841	—	0.9960	0	0	0	0.00%	14.43599
	MSEAT	1.7231	0	-0.0585	1	1	149303	71.10%	13.38152
	MSEAET	1.0511	1	-0.0583	1	1	125773	59.89%	13.38174
	MSEASVT	1.8276	0	-0.0568	1	1	148474	70.70%	13.38317
	MSESET	0.9365	1	0.9485	0	0	12637	6.02%	14.38846
20	Run Mean	0.9467	—	0.4559	1	0	0	0.00%	13.89592
	MSEAT	2.0492	0	-0.0572	1	1	130654	62.22%	13.38280
	MSEAET	1.0705	0	-0.0557	1	1	138050	65.74%	13.38434
	MSEASVT	1.7972	0	-0.0571	1	1	123737	58.92%	13.38286
	MSESET	0.8591	1	0.6846	0	0	59310	28.24%	14.12457
21	Run Mean	1.0232	—	-0.7268	0	0	0	0.00%	12.71323
	MSEAT	1.9310	0	-0.0577	1	1	172598	82.19%	13.38225
	MSEAET	0.9936	1	-0.0575	1	1	126381	60.18%	13.38247
	MSEASVT	1.9223	0	-0.0575	1	1	156483	74.52%	13.38251
	MSESET	0.8646	1	-0.3810	1	0	77315	36.82%	13.05904
22	Run Mean	1.0399	—	0.2320	1	1	0	0.00%	13.67199
	MSEAT	1.2966	0	-0.0571	1	1	127350	60.64%	13.38295
	MSEAET	1.3034	0	-0.0570	1	1	170933	81.40%	13.38304
	MSEASVT	1.1553	0	-0.0566	1	1	42537	20.26%	13.38343
	MSESET	0.6917	1	0.1160	1	1	20886	9.95%	13.55602
23	Run Mean	1.0911	—	0.5573	1	0	0	0.00%	13.99732
	MSEAT	4.2822	0	-0.0544	1	1	206507	98.34%	13.38561
	MSEAET	2.4026	0	-0.0561	1	1	142548	67.88%	13.38386
	MSEASVT	3.7282	0	-0.0654	1	1	202570	96.46%	13.37464
	MSESET	0.8019	1	0.5941	1	0	39059	18.60%	14.03407
24	Run Mean	0.8308	—	-0.3284	1	0	0	0.00%	13.11160
	MSEAT	0.9195	0	-0.0573	1	1	51104	24.34%	13.38267
	MSEAET	1.2357	0	-0.0562	1	1	144719	68.91%	13.38380
	MSEASVT	0.9312	0	-0.0574	1	1	41259	19.65%	13.38260
	MSESET	1.0097	0	0.4372	1	0	104969	49.99%	13.87720
25	Run Mean	1.0467	—	0.3418	1	0	0	0.00%	13.78184
	MSEAT	2.4424	0	-0.0552	1	1	205833	98.02%	13.38483
	MSEAET	2.1774	0	0.3225	1	0	119358	56.84%	13.76245
	MSEASVT	2.1128	0	-0.0573	1	1	120440	57.35%	13.38274
	MSESET	0.8652	1	0.8552	0	0	69957	33.31%	14.29522

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.1471	—	0.2198	0	0	0	0.00%	2.480844
	MSEAT	0.4918	0	0.0315	1	1	203747	97.02%	2.292456
	MSEAET	0.4586	0	0.1241	0	0	207415	98.77%	2.385092
	MSEASVT	0.2418	0	0.0468	1	0	175292	83.47%	2.307843
	MSESET	0.1198	1	0.2251	0	0	12815	6.10%	2.486126
2	Run Mean	0.1373	—	0.1391	0	0	0	0.00%	2.400066
	MSEAT	0.4742	0	0.0269	1	1	192291	91.57%	2.287869
	MSEAET	0.4320	0	0.0220	1	1	201211	95.81%	2.283047
	MSEASVT	0.4433	0	0.0265	1	1	182350	86.83%	2.287545
	MSESET	0.1320	1	0.1149	0	0	48346	23.02%	2.375853
3	Run Mean	0.1361	—	0.0316	1	1	0	0.00%	2.292619
	MSEAT	0.1698	0	0.0264	1	1	108762	51.79%	2.287424
	MSEAET	0.1521	0	-0.1044	1	0	144303	68.72%	2.156623
	MSEASVT	0.1307	1	0.0264	1	1	4523	2.15%	2.287410
	MSESET	0.1015	1	0.0289	1	1	81034	38.59%	2.289878
4	Run Mean	0.1017	—	-0.0020	1	1	0	0.00%	2.258998
	MSEAT	0.1247	0	0.0265	1	1	19174	9.13%	2.287478
	MSEAET	0.4894	0	0.0576	1	0	145661	69.36%	2.318572
	MSEASVT	0.1322	0	0.0263	1	1	12042	5.73%	2.287270
	MSESET	0.0810	1	-0.0042	1	1	16812	8.01%	2.256805
5	Run Mean	0.1043	—	0.0519	1	0	0	0.00%	2.312910
	MSEAT	0.1334	0	0.0264	1	1	29174	13.89%	2.287420
	MSEAET	0.3409	0	0.0218	1	1	148210	70.58%	2.282768
	MSEASVT	0.1334	0	0.0264	1	1	29174	13.89%	2.287420
	MSESET	0.1208	0	-0.0474	1	0	70995	33.81%	2.213557
6	Run Mean	0.1208	—	0.0263	1	1	0	0.00%	2.287320
	MSEAT	0.1410	0	0.0265	1	1	22828	10.87%	2.287482
	MSEAET	0.3458	0	0.0257	1	1	145541	69.31%	2.286685
	MSEASVT	0.1215	0	0.0265	1	1	16	0.01%	2.287481
	MSESET	0.1308	0	0.0797	1	0	101331	48.25%	2.340695
7	Run Mean	0.1689	—	0.1962	0	0	0	0.00%	2.457176
	MSEAT	0.4714	0	0.0250	1	1	197893	94.23%	2.286043
	MSEAET	0.4072	0	0.0270	1	1	204715	97.48%	2.287970
	MSEASVT	0.2739	0	0.0317	1	1	167803	79.91%	2.292669
	MSESET	0.1829	0	0.2912	0	0	93139	44.35%	2.552181
8	Run Mean	0.1491	—	-0.0213	1	1	0	0.00%	2.239743
	MSEAT	0.1782	0	0.0264	1	1	125891	59.95%	2.287374
	MSEAET	0.3782	0	0.0596	1	0	180283	85.85%	2.320576
	MSEASVT	0.1509	0	0.0264	1	1	17880	8.51%	2.287430
	MSESET	0.1317	1	0.0936	1	0	58484	27.85%	2.354631
9	Run Mean	0.0860	—	0.0531	1	0	0	0.00%	2.314064
	MSEAT	0.1253	0	0.0265	1	1	46172	21.99%	2.287466
	MSEAET	0.3243	0	-0.0442	1	1	151436	72.11%	2.216817
	MSEASVT	0.1129	0	0.0265	1	1	11040	5.26%	2.287513
	MSESET	0.0745	1	0.0036	1	1	61517	29.29%	2.264648

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.0860	—	0.0531	1	0	0	0.00%	2.314064
	MSEAT	0.1253	0	0.0265	1	1	46172	21.99%	2.287466
	MSEAET	0.3243	0	-0.0442	1	1	151436	72.11%	2.216817
	MSEASVT	0.1129	0	0.0265	1	1	11040	5.26%	2.287513
	MSESET	0.0745	1	0.0036	1	1	61517	29.29%	2.264648
11	Run Mean	0.1336	—	0.0142	1	1	0	0.00%	2.275225
	MSEAT	0.1555	0	0.0264	1	1	30438	14.49%	2.287411
	MSEAET	0.2407	0	-0.0387	1	1	131026	62.39%	2.222289
	MSEASVT	0.1454	0	0.0264	1	1	7728	3.68%	2.287416
	MSESET	0.1624	0	0.0735	1	0	98008	46.67%	2.334479
12	Run Mean	0.1032	—	-0.0392	1	1	0	0.00%	2.221827
	MSEAT	0.2509	0	0.0263	1	1	153144	72.93%	2.287315
	MSEAET	0.2368	0	-0.1311	0	0	189107	90.05%	2.129942
	MSEASVT	0.2091	0	0.0264	1	1	148456	70.69%	2.287404
	MSESET	0.1091	0	-0.1377	0	0	93117	44.34%	2.123265
13	Run Mean	0.1288	—	0.1309	0	0	0	0.00%	2.391876
	MSEAT	0.1776	0	0.0265	1	1	111719	53.20%	2.287476
	MSEAET	0.3192	0	0.1119	1	0	176475	84.04%	2.372933
	MSEASVT	0.1722	0	0.0265	1	1	110296	52.52%	2.287456
	MSESET	0.1286	1	0.0555	1	0	103443	49.26%	2.316483
14	Run Mean	0.1406	—	0.0130	1	1	0	0.00%	2.273952
	MSEAT	0.1286	1	0.0265	1	1	19413	9.24%	2.287476
	MSEAET	0.4189	0	0.0219	1	1	146061	69.55%	2.282864
	MSEASVT	0.1311	1	0.0261	1	1	5385	2.56%	2.287112
	MSESET	0.0949	1	0.0737	1	0	45511	21.67%	2.334708
15	Run Mean	0.1237	—	0.1369	0	0	0	0.00%	2.397907
	MSEAT	0.4178	0	0.0266	1	1	185228	88.20%	2.287555
	MSEAET	0.5308	0	0.0776	1	0	199995	95.24%	2.338563
	MSEASVT	0.4178	0	0.0266	1	1	185228	88.20%	2.287555
	MSESET	0.0756	1	0.1234	0	0	39044	18.59%	2.384376
16	Run Mean	0.1580	—	-0.0238	1	1	0	0.00%	2.237229
	MSEAT	0.1812	0	0.0134	1	1	59421	28.30%	2.274402
	MSEAET	0.1453	1	-0.1348	0	0	102996	49.05%	2.126207
	MSEASVT	0.1812	0	0.0134	1	1	59421	28.30%	2.274402
	MSESET	0.1716	0	-0.1727	0	0	100171	47.70%	2.088319
17	Run Mean	0.1570	—	0.0908	1	0	0	0.00%	2.351807
	MSEAT	0.2243	0	0.0264	1	1	165665	78.89%	2.287433
	MSEAET	0.3772	0	0.0269	1	1	194071	92.41%	2.287897
	MSEASVT	0.2374	0	0.0265	1	1	164038	78.11%	2.287469
	MSESET	0.1724	0	0.2131	0	0	101077	48.13%	2.474122
18	Run Mean	0.1347	—	-0.0690	1	0	0	0.00%	2.192037
	MSEAT	0.2682	0	-0.0354	1	1	176450	84.02%	2.225610
	MSEAET	0.1805	0	-0.1677	0	0	184631	87.92%	2.093282
	MSEASVT	0.2682	0	-0.0354	1	1	176450	84.02%	2.225610
	MSESET	0.0941	1	-0.0991	1	0	54684	26.04%	2.161918

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
	Run Mean	0.1074	—	0.0218	1	1	0	0.00%	2.282784
	MSEAT	0.1136	0	0.0265	1	1	3480	1.66%	2.287476
19	MSEAET	0.3294	0	0.1439	0	0	125093	59.57%	2.404912
	MSEASVT	0.1101	0	0.0265	1	1	2184	1.04%	2.287485
	MSESET	0.0900	1	0.0293	1	1	38360	18.27%	2.290307
	Run Mean	0.1144	—	-0.0096	1	1	0	0.00%	2.251363
	MSEAT	0.3360	0	0.0280	1	1	184257	87.74%	2.288995
20	MSEAET	0.4023	0	0.0616	1	0	199348	94.93%	2.322562
	MSEASVT	0.3362	0	0.0279	1	1	184256	87.74%	2.288859
	MSESET	0.1283	0	-0.1096	1	0	91740	43.69%	2.151402
	Run Mean	0.1637	—	0.0137	1	1	0	0.00%	2.274745
	MSEAT	0.1554	1	0.0265	1	1	53278	25.37%	2.287496
21	MSEAET	0.3151	0	0.0158	1	1	150983	71.90%	2.276758
	MSEASVT	0.1529	1	0.0265	1	1	4791	2.28%	2.287521
	MSESET	0.1649	0	0.1100	1	0	103592	49.33%	2.370996
	Run Mean	0.1589	—	0.0380	1	1	0	0.00%	2.299047
	MSEAT	0.3502	0	0.0262	1	1	172310	82.05%	2.287186
22	MSEAET	0.2714	0	-0.0746	1	0	185252	88.22%	2.186387
	MSEASVT	0.1755	0	0.0264	1	1	50084	23.85%	2.287410
	MSESET	0.1734	0	-0.0232	1	1	68975	32.85%	2.237764
	Run Mean	0.1131	—	0.0512	1	0	0	0.00%	2.312200
	MSEAT	0.1386	0	0.0264	1	1	68630	32.68%	2.287444
23	MSEAET	0.1592	0	-0.1067	1	0	96816	46.10%	2.154332
	MSEASVT	0.1335	0	0.0265	1	1	56851	27.07%	2.287461
	MSESET	0.0821	1	0.0294	1	1	36369	17.32%	2.290447
	Run Mean	0.0814	—	-0.0605	1	0	0	0.00%	2.200511
	MSEAT	0.3071	0	0.0265	1	1	170133	81.02%	2.287510
24	MSEAET	0.5188	0	0.0105	1	1	195970	93.32%	2.271510
	MSEASVT	0.2338	0	0.0265	1	1	156253	74.41%	2.287469
	MSESET	0.0813	1	-0.0775	1	0	71021	33.82%	2.183463
	Run Mean	0.1644	—	0.1593	0	0	0	0.00%	2.420261
	MSEAT	0.2862	0	0.0263	1	1	158808	75.62%	2.287343
25	MSEAET	0.4439	0	0.0131	1	1	186083	88.61%	2.274097
	MSEASVT	0.2949	0	0.0263	1	1	158632	75.54%	2.287275
	MSESET	0.1056	1	0.1577	0	0	36772	17.51%	2.418660

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	0.1498	—	0.2176	0	0	0	0.00%	2.47857
	MSEAT	0.4278	0	0.0305	1	1	202103	96.24%	2.29154
	MSEAET	0.3971	0	0.0962	1	0	206867	98.51%	2.35723
	MSEASVT	0.2418	0	0.0467	1	0	175292	83.47%	2.30771
	MSESET	0.1225	1	0.2251	0	0	12923	6.15%	2.48611
2	Run Mean	0.1370	—	0.1391	0	0	0	0.00%	2.40007
	MSEAT	0.5382	0	0.0269	1	1	187042	89.07%	2.28788
	MSEAET	0.4289	0	0.0250	1	1	199045	94.78%	2.28601
	MSEASVT	0.4433	0	0.0265	1	1	182350	86.83%	2.28751
	MSESET	0.1320	1	0.1149	0	0	48346	23.02%	2.37586
3	Run Mean	0.1361	—	0.0317	1	1	0	0.00%	2.29272
	MSEAT	0.1410	0	0.0265	1	1	17013	8.10%	2.28748
	MSEAET	0.1376	0	-0.1044	1	0	113720	54.15%	2.15664
	MSEASVT	0.1307	1	0.0264	1	1	4523	2.15%	2.28740
	MSESET	0.1015	1	0.0289	1	1	16801	8.00%	2.28986
4	Run Mean	0.1509	—	0.0850	1	0	0	0.00%	2.34598
	MSEAT	0.4472	0	0.0345	1	1	200279	95.37%	2.29547
	MSEAET	0.4992	0	0.1866	0	0	206246	98.21%	2.44755
	MSEASVT	0.2038	0	0.0265	1	1	129280	61.56%	2.28752
	MSESET	0.1430	1	0.1877	0	0	98827	47.06%	2.44865
5	Run Mean	0.1053	—	0.0495	1	0	0	0.00%	2.31049
	MSEAT	0.1334	0	0.0264	1	1	29174	13.89%	2.28743
	MSEAET	0.3442	0	0.0290	1	1	148209	70.58%	2.28998
	MSEASVT	0.1334	0	0.0264	1	1	29174	13.89%	2.28743
	MSESET	0.1208	0	-0.0474	1	0	70995	33.81%	2.21361
6	Run Mean	0.1208	—	0.0263	1	1	0	0.00%	2.28732
	MSEAT	0.1219	0	0.0265	1	1	17825	8.49%	2.28748
	MSEAET	0.3584	0	0.0268	1	1	143874	68.51%	2.28780
	MSEASVT	0.1214	0	0.0265	1	1	13	0.01%	2.28747
	MSESET	0.1308	0	0.0797	1	0	101332	48.25%	2.34074
7	Run Mean	0.1417	—	0.1572	0	0	0	0.00%	2.41823
	MSEAT	0.3608	0	0.0251	1	1	197330	93.97%	2.28614
	MSEAET	0.3750	0	0.0388	1	1	204836	97.54%	2.29975
	MSEASVT	0.3445	0	0.0259	1	1	196146	93.40%	2.28690
	MSESET	0.1513	0	0.2438	0	0	88941	42.35%	2.50480
8	Run Mean	0.1493	—	-0.0212	1	1	0	0.00%	2.23977
	MSEAT	0.1491	1	0.0265	1	1	22239	10.59%	2.28748
	MSEAET	0.3123	0	0.0567	1	0	145691	69.38%	2.31772
	MSEASVT	0.1508	0	0.0265	1	1	17871	8.51%	2.28746
	MSESET	0.1316	1	0.0938	1	0	58484	27.85%	2.35481
9	Run Mean	0.0869	—	0.0530	1	0	0	0.00%	2.31402
	MSEAT	0.1136	0	0.0264	1	1	77711	37.01%	2.28745
	MSEAET	0.3105	0	-0.0451	1	1	161934	77.11%	2.21585
	MSEASVT	0.1123	0	0.0265	1	1	11037	5.26%	2.28752
	MSESET	0.0745	1	0.0036	1	1	61517	29.29%	2.26461

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	0.1437	—	-0.0150	1	1	0	0.00%	2.24603
	MSEAT	0.2303	0	0.0263	1	1	123409	58.77%	2.28730
	MSEAET	0.1844	0	-0.1184	0	0	149401	71.14%	2.14256
	MSEASVT	0.2367	0	0.0262	1	1	122871	58.51%	2.28724
	MSESET	0.1147	1	-0.0287	1	1	23072	10.99%	2.23231
11	Run Mean	0.1278	—	-0.0506	1	0	0	0.00%	2.21043
	MSEAT	0.6417	0	0.0220	1	1	208150	99.12%	2.28296
	MSEAET	0.5417	0	0.1619	0	0	207804	98.95%	2.42294
	MSEASVT	0.1637	0	0.0207	1	1	74160	35.31%	2.28165
	MSESET	0.1013	1	-0.0559	1	0	43006	20.48%	2.20507
12	Run Mean	0.1025	—	-0.0395	1	1	0	0.00%	2.22153
	MSEAT	0.2509	0	0.0263	1	1	153144	72.93%	2.28733
	MSEAET	0.2425	0	-0.1144	0	0	189104	90.05%	2.14656
	MSEASVT	0.2091	0	0.0264	1	1	148456	70.69%	2.28740
	MSESET	0.1090	0	-0.1377	0	0	93117	44.34%	2.12327
13	Run Mean	0.1279	—	0.1329	0	0	0	0.00%	2.39385
	MSEAT	0.1780	0	0.0264	1	1	111718	53.20%	2.28742
	MSEAET	0.3992	0	0.0896	1	0	176937	84.26%	2.35058
	MSEASVT	0.1723	0	0.0265	1	1	110297	52.52%	2.28747
	MSESET	0.1283	0	0.0555	1	0	103443	49.26%	2.31650
14	Run Mean	0.1406	—	0.0130	1	1	0	0.00%	2.27399
	MSEAT	0.1286	1	0.0265	1	1	19413	9.24%	2.28748
	MSEAET	0.4189	0	0.0219	1	1	146061	69.55%	2.28286
	MSEASVT	0.1311	1	0.0261	1	1	5385	2.56%	2.28711
	MSESET	0.0949	1	0.0737	1	0	45511	21.67%	2.33471
15	Run Mean	0.1223	—	0.1367	0	0	0	0.00%	2.39767
	MSEAT	0.4736	0	0.0275	1	1	194874	92.80%	2.28851
	MSEAET	0.4826	0	0.1133	0	0	203196	96.76%	2.37428
	MSEASVT	0.2017	0	0.0414	1	1	136612	65.05%	2.30244
	MSESET	0.0756	1	0.1232	0	0	39047	18.59%	2.38416
16	Run Mean	0.1580	—	-0.0237	1	1	0	0.00%	2.23728
	MSEAT	0.1812	0	0.0134	1	1	59421	28.30%	2.27440
	MSEAET	0.1453	1	-0.1348	0	0	102996	49.05%	2.12621
	MSEASVT	0.1812	0	0.0134	1	1	59421	28.30%	2.27440
	MSESET	0.1716	0	-0.1727	0	0	100171	47.70%	2.08832
17	Run Mean	0.1566	—	0.0907	1	0	0	0.00%	2.35174
	MSEAT	0.2363	0	0.0264	1	1	166659	79.36%	2.28739
	MSEAET	0.3250	0	0.0268	1	1	194243	92.50%	2.28785
	MSEASVT	0.2378	0	0.0262	1	1	164038	78.11%	2.28715
	MSESET	0.1724	0	0.2131	0	0	101078	48.13%	2.47409
18	Run Mean	0.1358	—	-0.0689	1	0	0	0.00%	2.19206
	MSEAT	0.2682	0	-0.0353	1	1	176450	84.02%	2.22566
	MSEAET	0.1915	0	-0.2413	0	0	195814	93.24%	2.01972
	MSEASVT	0.2682	0	-0.0353	1	1		0.00%	2.22566
	MSESET	0.0941	1	-0.0991	1	0		0.00%	2.16194

Micro Analysis of Results

Model: U / Ln / 3 Batch Means w / Empty & Idle

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
19	Run Mean	0.1071	—	0.0218	1	1	0	0.00%	2.28282
	MSEAT	0.1266	0	0.0265	1	1	29766	14.17%	2.28748
	MSEAET	0.3317	0	0.1394	0	0	133854	63.74%	2.40042
	MSEASVT	0.1104	0	0.0264	1	1	2165	1.03%	2.28741
	MSESET	0.0900	1	0.0294	1	1	38360	18.27%	2.29041
20	Run Mean	0.1362	—	-0.0618	1	0	0	0.00%	2.19917
	MSEAT	0.5784	0	0.0271	1	1	191005	90.95%	2.28811
	MSEAET	0.5167	0	0.0790	1	0	203327	96.82%	2.34001
	MSEASVT	0.4145	0	0.0242	1	1	184035	87.64%	2.28519
	MSESET	0.1564	0	-0.1376	0	0	91751	43.69%	2.12338
21	Run Mean	0.1636	—	0.0137	1	1	0	0.00%	2.27467
	MSEAT	0.1738	0	0.0265	1	1	27816	13.25%	2.28748
	MSEAET	0.3077	0	0.0149	1	1	142496	67.86%	2.27590
	MSEASVT	0.1529	1	0.0265	1	1	4792	2.28%	2.28752
	MSESET	0.1649	0	0.1100	1	0	103592	49.33%	2.37098
22	Run Mean	0.1586	—	0.0383	1	1	0	0.00%	2.29925
	MSEAT	0.3504	0	0.0262	1	1	172313	82.05%	2.28718
	MSEAET	0.2715	0	-0.0749	1	0	185257	88.22%	2.18615
	MSEASVT	0.1755	0	0.0264	1	1	50084	23.85%	2.28738
	MSESET	0.1735	0	-0.0233	1	1	68975	32.85%	2.23772
23	Run Mean	0.1131	—	0.0512	1	0	0	0.00%	2.31224
	MSEAT	0.1345	0	0.0264	1	1	66960	31.89%	2.28743
	MSEAET	0.1621	0	-0.1066	1	0	96259	45.84%	2.15436
	MSEASVT	0.1329	0	0.0264	1	1	56851	27.07%	2.28739
	MSESET	0.0821	1	0.0295	1	1	36369	17.32%	2.29047
24	Run Mean	0.0805	—	-0.0607	1	0	0	0.00%	2.20029
	MSEAT	0.2866	0	0.0266	1	1	176319	83.96%	2.28759
	MSEAET	0.6686	0	-0.0075	1	1	198061	94.31%	2.25352
	MSEASVT	0.2340	0	0.0265	1	1	156259	74.41%	2.28750
	MSESET	0.0814	0	-0.0776	1	0	71021	33.82%	2.18341
25	Run Mean	0.1600	—	0.1550	0	0	0	0.00%	2.41599
	MSEAT	0.3498	0	0.0264	1	1	170043	80.97%	2.28737
	MSEAET	0.4426	0	0.0139	1	1	189278	90.13%	2.27486
	MSEASVT	0.2951	0	0.0263	1	1	158631	75.54%	2.28730
	MSESET	0.1047	1	0.1567	0	0	36715	17.48%	2.41775

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	16.3975	—	-13.0664	0	0	0	0.00%	79.72964
	MSEAT	30.3772	0	-2.7936	1	1	102928	49.01%	90.00241
	MSEAET	30.3772	0	-2.7936	1	1	102928	49.01%	90.00241
	MSEASVT	14.7683	1	-12.1421	0	0	11125	5.30%	80.65388
	MSEASVT(A)	20.8613	0	-0.1258	1	1	114759	54.65%	92.67021
2	Run Mean	9.0668	—	-20.2713	0	0	0	0.00%	72.52470
	MSEAT	19.9487	0	-20.3534	0	0	82173	39.13%	72.44262
	MSEAET	15.9417	0	-20.2907	0	0	64578	30.75%	72.50531
	MSEASVT	20.9646	0	-19.8920	0	0	78925	37.58%	72.90404
	MSEASVT(A)	45.9434	0	-0.1967	1	1	203817	97.06%	92.59934
3	Run Mean	13.7412	—	-15.6804	0	0	0	0.00%	77.11559
	MSEAT	16.9458	0	-14.0155	0	0	54927	26.16%	78.78046
	MSEAET	11.5777	1	-16.3665	0	0	79668	37.94%	76.42945
	MSEASVT	13.2669	1	-11.3908	0	0	18765	8.94%	81.40521
	MSEASVT(A)	8.2947	1	-0.4349	1	1	209899	99.95%	92.36109
4	Run Mean	30.6707	—	2.1098	1	1	0	0.00%	94.90581
	MSEAT	12.9280	1	-23.1112	0	0	99197	47.24%	69.68478
	MSEAET	13.2108	1	-22.9550	0	0	99210	47.24%	69.84098
	MSEASVT	13.0063	1	-22.7515	0	0	97900	46.62%	70.04446
	MSEASVT(A)	30.7786	0	-0.1245	1	1	11894	5.66%	92.67146
5	Run Mean	18.9146	—	2.5704	1	1	0	0.00%	95.36645
	MSEAT	23.7642	0	10.8168	0	0	50868	24.22%	103.61281
	MSEAET	16.1074	1	11.6512	0	0	72443	34.50%	104.44722
	MSEASVT	23.7695	0	10.8121	0	0	50859	24.22%	103.60805
	MSEASVT(A)	60.3173	0	-0.1510	1	1	184358	87.79%	92.64497
6	Run Mean	21.3403	—	8.9929	1	0	0	0.00%	101.78889
	MSEAT	21.1579	1	5.6131	1	0	17991	8.57%	98.40912
	MSEAET	12.7913	1	8.7630	1	0	59896	28.52%	101.55897
	MSEASVT	20.7997	1	5.4534	1	0	12428	5.92%	98.24939
	MSEASVT(A)	31.0244	0	-0.1245	1	1	128621	61.25%	92.67154
7	Run Mean	12.1379	—	-7.2998	1	0	0	0.00%	85.49617
	MSEAT	15.1412	0	-11.3568	0	0	52618	25.06%	81.43921
	MSEAET	11.2017	1	-11.1857	0	0	64288	30.61%	81.61026
	MSEASVT	15.1412	0	-11.3568	0	0	52618	25.06%	81.43921
	MSEASVT(A)	60.1977	0	-0.1503	1	1	189075	90.04%	92.64575
8	Run Mean	13.6649	—	-9.6070	0	0	0	0.00%	83.18896
	MSEAT	14.9471	0	-7.9939	1	0	11715	5.58%	84.80215
	MSEAET	11.5460	1	-10.1201	0	0	55711	26.53%	82.67591
	MSEASVT	14.9457	0	-7.9952	1	0	11712	5.58%	84.80081
	MSEASVT(A)	48.5668	0	-0.1261	1	1	177660	84.60%	92.66985
9	Run Mean	19.8557	—	-5.5940	1	0	0	0.00%	87.20196
	MSEAT	33.4349	0	7.0606	1	0	104844	49.93%	99.85664
	MSEAET	27.0965	0	3.8205	1	1	74875	35.65%	96.61653
	MSEASVT	26.5878	0	-0.4230	1	1	42999	20.48%	92.37305
	MSEASVT(A)	31.2750	0	-0.1152	1	1	87438	41.64%	92.68079

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	14.8659	—	-7.0132	1	0	0	0.00%	85.78278
	MSEAT	21.1973	0	1.6630	1	1	54263	25.84%	94.45898
	MSEAET	16.5587	0	0.4207	1	1	51619	24.58%	93.21666
	MSEASVT	21.1930	0	1.6605	1	1	54256	25.84%	94.45647
	MSEASVT(A)	21.7214	0	-0.1226	1	1	49405	23.53%	92.67335
11	Run Mean	11.0682	—	-18.4230	0	0	0	0.00%	74.37304
	MSEAT	13.4621	0	-14.9467	0	0	100086	47.66%	77.84930
	MSEAET	9.6957	1	-14.8020	0	0	75694	36.04%	77.99398
	MSEASVT	11.1922	0	-12.2648	0	0	63196	30.09%	80.53123
	MSEASVT(A)	19.2584	0	-0.1271	1	1	182556	86.93%	92.66891
12	Run Mean	20.7730	—	9.1233	1	0	0	0.00%	101.91935
	MSEAT	28.7265	0	1.0610	1	1	94587	45.04%	93.85702
	MSEAET	23.7899	0	2.3216	1	1	76789	36.57%	95.11761
	MSEASVT	28.7301	0	1.0637	1	1	94585	45.04%	93.85969
	MSEASVT(A)	28.2330	0	-0.1188	1	1	95625	45.54%	92.67722
13	Run Mean	31.8309	—	27.1510	0	0	0	0.00%	119.94705
	MSEAT	35.1719	0	30.9316	0	0	13212	6.29%	123.72765
	MSEAET	30.4833	1	43.7834	0	0	55112	26.24%	136.57944
	MSEASVT	35.1673	0	30.9251	0	0	13200	6.29%	123.72109
	MSEASVT(A)	54.0511	0	-0.1394	1	1	207911	99.01%	92.65663
14	Run Mean	38.0373	—	3.2874	1	1	0	0.00%	96.08338
	MSEAT	59.8804	0	28.8426	0	0	98965	47.13%	121.63862
	MSEAET	50.5145	0	27.5821	0	0	95496	45.47%	120.37814
	MSEASVT	56.5589	0	27.9361	0	0	96597	46.00%	120.73214
	MSEASVT(A)	30.2117	1	-0.1222	1	1	158071	75.27%	92.67375
15	Run Mean	17.1068	—	-20.3886	0	0	0	0.00%	72.40738
	MSEAT	24.2635	0	-16.9311	0	0	40456	19.26%	75.86487
	MSEAET	8.4953	1	-21.5672	0	0	51147	24.36%	71.22884
	MSEASVT	24.2632	0	-16.9316	0	0	40454	19.26%	75.86435
	MSEASVT(A)	24.3109	0	-16.7863	0	0	41015	19.53%	76.00971
16	Run Mean	13.2051	—	-6.1371	1	0	0	0.00%	86.65886
	MSEAT	13.0048	1	-5.8570	1	0	989	0.47%	86.93900
	MSEAET	11.2708	1	-7.1454	1	0	55559	26.46%	85.65058
	MSEASVT	13.0048	1	-5.8574	1	0	987	0.47%	86.93864
	MSEASVT(A)	16.1903	0	-1.3218	1	1	40641	19.35%	91.47425
17	Run Mean	24.2990	—	7.4744	1	0	0	0.00%	100.27040
	MSEAT	26.0949	0	10.0925	0	0	15114	7.20%	102.88850
	MSEAET	24.0369	1	12.3485	0	0	57902	27.57%	105.14451
	MSEASVT	41.9438	0	18.8379	0	0	104999	50.00%	111.63394
	MSEASVT(A)	48.6713	0	-0.1127	1	1	170607	81.24%	92.68329
18	Run Mean	34.5779	—	3.4073	1	1	0	0.00%	96.20330
	MSEAT	49.8511	0	21.3574	0	0	86225	41.06%	114.15336
	MSEAET	46.3446	0	18.6686	0	0	75058	35.74%	111.46462
	MSEASVT	49.8420	0	21.3011	0	0	86076	40.99%	114.09714
	MSEASVT(A)	62.8257	0	-0.1339	1	1	190381	90.66%	92.66214

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Initial Q Length (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
19	Run Mean	11.9939	—	-21.3719	0	0	0	0.00%	71.42407
	MSEAT	16.3837	0	-19.2812	0	0	14873	7.08%	73.51476
	MSEAET	15.9897	0	-19.5367	0	0	15264	7.27%	73.25933
	MSEASVT	16.3837	0	-19.2812	0	0	14873	7.08%	73.51476
	MSEASVT(A)	54.1816	0	-0.0580	1	1	205103	97.67%	92.73797
20	Run Mean	11.2896	—	-14.2757	0	0	0	0.00%	78.52031
	MSEAT	13.7765	0	-14.6111	0	0	95958	45.69%	78.18488
	MSEAET	10.4280	1	-14.9179	0	0	70924	33.77%	77.87807
	MSEASVT	13.8221	0	-14.6235	0	0	95940	45.69%	78.17251
	MSEASVT(A)	41.9924	0	-0.1153	1	1	200122	95.30%	92.68071
21	Run Mean	19.0474	—	-7.7166	1	0	0	0.00%	85.07941
	MSEAT	16.2572	1	-5.8417	1	0	16051	7.64%	86.95430
	MSEAET	15.4412	1	-9.3914	0	0	56495	26.90%	83.40456
	MSEASVT	16.4422	1	-5.7981	1	0	14514	6.91%	86.99791
	MSEASVT(A)	31.8574	0	-0.1253	1	1	147927	70.44%	92.67067
22	Run Mean	9.2136	—	-17.8338	0	0	0	0.00%	74.96224
	MSEAT	13.2664	0	-15.7138	0	0	104984	49.99%	77.08222
	MSEAET	9.3521	0	-15.4601	0	0	76264	36.32%	77.33591
	MSEASVT	12.5161	0	-12.7196	0	0	38171	18.18%	80.07636
	MSEASVT(A)	13.6068	0	-0.1847	1	1	208675	99.37%	92.61132
23	Run Mean	20.8074	—	-15.1490	0	0	0	0.00%	77.64700
	MSEAT	21.0726	0	-14.9278	0	0	734	0.35%	77.86817
	MSEAET	9.6576	1	-22.2432	0	0	39707	18.91%	70.55280
	MSEASVT	21.0716	0	-14.9281	0	0	733	0.35%	77.86788
	MSEASVT(A)	37.8561	0	-0.1293	1	1	202760	96.55%	92.66667
24	Run Mean	16.6097	—	-10.3669	0	0	0	0.00%	82.42915
	MSEAT	16.2223	1	-9.9459	0	0	1459	0.69%	82.85010
	MSEAET	9.6408	1	-11.9410	0	0	45215	21.53%	80.85503
	MSEASVT	16.2213	1	-9.9464	0	0	1458	0.69%	82.84964
	MSEASVT(A)	40.6442	0	-0.1328	1	1	188423	89.73%	92.66320
25	Run Mean	13.0650	—	0.5478	1	1	0	0.00%	93.34381
	MSEAT	13.5100	0	2.9430	1	1	9373	4.46%	95.73899
	MSEAET	8.7463	1	3.1126	1	1	52497	25.00%	95.90862
	MSEASVT	13.5134	0	2.9403	1	1	9356	4.46%	95.73634
	MSEASVT(A)	17.5602	0	-0.1211	1	1	57959	27.60%	92.67494

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
1	Run Mean	17.0265	—	-14.1765	0	0	0	0.00%	78.61954
	MSEAT	29.1068	0	-3.4821	1	1	102905	49.00%	89.31390
	MSEAET	29.1068	0	-3.4821	1	1	102905	49.00%	89.31390
	MSEASVT	15.3424	1	-12.8283	0	0	11127	5.30%	79.96767
	MSEASVT(A)	21.0946	0	-0.1250	1	1	115032	54.78%	92.67097
2	Run Mean	9.0747	—	-20.2716	0	0	0	0.00%	72.52435
	MSEAT	19.9485	0	-20.3540	0	0	82173	39.13%	72.44198
	MSEAET	16.0412	0	-20.2995	0	0	64826	30.87%	72.49648
	MSEASVT	20.9645	0	-19.8926	0	0	78925	37.58%	72.90342
	MSEASVT(A)	45.9272	0	-0.2050	1	1	203819	97.06%	92.59101
3	Run Mean	12.7605	—	-16.6315	0	0	0	0.00%	76.16448
	MSEAT	13.9910	0	-13.3424	0	0	36160	17.22%	79.45362
	MSEAET	9.4979	1	-16.9936	0	0	73483	34.99%	75.80243
	MSEASVT	13.9872	0	-13.3445	0	0	36149	17.21%	79.45153
	MSEASVT(A)	12.1333	1	-12.5276	0	0	19800	9.43%	80.26845
4	Run Mean	32.4012	—	2.2663	1	1	0	0.00%	95.06229
	MSEAT	14.1266	1	-24.8472	0	0	98965	47.13%	67.94879
	MSEAET	14.1266	1	-24.7217	0	0	99086	47.18%	68.07435
	MSEASVT	14.0352	1	-24.5797	0	0	97993	46.66%	68.21631
	MSEASVT(A)	33.6985	0	-0.1246	1	1	11837	5.64%	92.67140
5	Run Mean	19.0495	—	3.8521	1	1	0	0.00%	96.64808
	MSEAT	23.5003	0	13.0277	0	0	53335	25.40%	105.82368
	MSEAET	16.2706	1	14.1205	0	0	73408	34.96%	106.91652
	MSEASVT	23.8668	0	12.9162	0	0	52081	24.80%	105.71217
	MSEASVT(A)	59.1004	0	-0.1355	1	1	185100	88.14%	92.66052
6	Run Mean	22.4977	—	7.0383	1	0	0	0.00%	99.83434
	MSEAT	28.6372	0	9.0022	1	0	103644	49.35%	101.79819
	MSEAET	25.3678	0	8.9829	1	0	93767	44.65%	101.77893
	MSEASVT	21.7826	1	4.7481	1	0	10765	5.13%	97.54405
	MSEASVT(A)	31.3014	0	-0.1247	1	1	127696	60.81%	92.67134
7	Run Mean	11.0961	—	-7.7698	1	0	0	0.00%	85.02616
	MSEAT	19.1367	0	-11.2014	0	0	57191	27.23%	81.59456
	MSEAET	11.3192	0	-10.8534	0	0	66654	31.74%	81.94255
	MSEASVT	14.6692	0	-10.8610	0	0	52383	24.94%	81.93503
	MSEASVT(A)	55.9495	0	-0.1373	1	1	189594	90.28%	92.65871
8	Run Mean	14.8261	—	-9.2032	1	0	0	0.00%	83.59275
	MSEAT	17.3250	0	-7.7317	1	0	20818	9.91%	85.06428
	MSEAET	13.7468	1	-9.4958	0	0	55144	26.26%	83.30023
	MSEASVT	16.4695	0	-7.4004	1	0	12193	5.81%	85.39562
	MSEASVT(A)	34.5210	0	-0.1429	1	1	155391	74.00%	92.65309
9	Run Mean	20.9923	—	-4.5776	1	1	0	0.00%	88.21835
	MSEAT	36.5907	0	8.8400	1	0	104713	49.86%	101.63601
	MSEAET	31.4261	0	5.6936	1	0	80158	38.17%	98.48957
	MSEASVT	28.0374	0	1.1034	1	1	43009	20.48%	93.89945
	MSEASVT(A)	25.7988	0	-0.1211	1	1	32140	15.30%	92.67488

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
10	Run Mean	14.3840	—	-8.1110	1	0	0	0.00%	84.68498
	MSEAT	19.5070	0	1.0111	1	1	53833	25.63%	93.80708
	MSEAET	15.2951	0	0.0218	1	1	44356	21.12%	92.81783
	MSEASVT	19.4979	0	1.0065	1	1	53828	25.63%	93.80251
	MSEASVT(A)	19.7998	0	-0.1216	1	1	50643	24.12%	92.67436
11	Run Mean	12.5535	—	-17.5480	0	0	0	0.00%	75.24798
	MSEAT	11.7479	1	-11.2695	0	0	65041	30.97%	81.52646
	MSEAET	6.5335	1	-13.5311	0	0	61620	29.34%	79.26495
	MSEASVT	12.0122	1	-10.8995	0	0	60024	28.58%	81.89649
	MSEASVT(A)	28.2493	0	-0.1297	1	1	193865	92.32%	92.66633
12	Run Mean	20.9346	—	7.3302	1	0	0	0.00%	100.12624
	MSEAT	28.9138	0	1.1266	1	1	94597	45.05%	93.92260
	MSEAET	23.8925	0	2.7653	1	1	67029	31.92%	95.56128
	MSEASVT	28.9171	0	1.1291	1	1	94595	45.05%	93.92510
	MSEASVT(A)	28.4311	0	-0.1141	1	1	95793	45.62%	92.68187
13	Run Mean	31.9506	—	27.0474	0	0	0	0.00%	119.84338
	MSEAT	35.4786	0	30.4679	0	0	13201	6.29%	123.26385
	MSEAET	29.6183	1	33.2434	0	0	21439	10.21%	126.03944
	MSEASVT	35.4721	0	30.4587	0	0	13179	6.28%	123.25465
	MSEASVT(A)	51.5360	0	-0.2562	1	1	207980	99.04%	92.53979
14	Run Mean	37.6545	—	4.0127	1	1	0	0.00%	96.80865
	MSEAT	63.1196	0	30.4788	0	0	101769	48.46%	123.27480
	MSEAET	49.8608	0	28.8041	0	0	97314	46.34%	121.60006
	MSEASVT	55.5412	0	28.4948	0	0	96528	45.97%	121.29076
	MSEASVT(A)	25.0282	1	-0.1200	1	1	157747	75.12%	92.67603
15	Run Mean	18.3191	—	-21.2423	0	0	0	0.00%	71.55373
	MSEAT	23.3618	0	-18.5318	0	0	38696	18.43%	74.26422
	MSEAET	7.8530	1	-23.5309	0	0	49318	23.48%	69.26508
	MSEASVT	23.3552	0	-18.5324	0	0	38694	18.43%	74.26365
	MSEASVT(A)	25.8541	0	-17.8586	0	0	41028	19.54%	74.93743
16	Run Mean	13.9314	—	-5.8234	1	0	0	0.00%	86.97258
	MSEAT	13.8909	1	-5.6673	1	0	583	0.28%	87.12875
	MSEAET	11.0606	1	-6.2481	1	0	42754	20.36%	86.54791
	MSEASVT	13.8861	1	-5.6677	1	0	578	0.28%	87.12826
	MSEASVT(A)	16.9101	0	-1.0929	1	1	40630	19.35%	91.70313
17	Run Mean	24.2774	—	7.4778	1	0	0	0.00%	100.27380
	MSEAT	26.0919	0	10.0970	0	0	15114	7.20%	102.89298
	MSEAET	25.1975	0	12.9911	0	0	62016	29.53%	105.78706
	MSEASVT	26.0968	0	10.0937	0	0	15110	7.20%	102.88971
	MSEASVT(A)	48.6556	0	-0.1131	1	1	170610	81.24%	92.68289
18	Run Mean	33.8037	—	3.8427	1	1	0	0.00%	96.63875
	MSEAT	48.6710	0	20.8432	0	0	85719	40.82%	113.63919
	MSEAET	42.3974	0	17.4802	0	0	68752	32.74%	110.27620
	MSEASVT	48.6675	0	20.8177	0	0	85675	40.80%	113.61368
	MSEASVT(A)	63.5557	0	-0.1567	1	1	190321	90.63%	92.63935

Micro Analysis of Results

Model: E / E / 4 Batch Means w / Empty & Idle (Large Expected QL)

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Trunc Pt.	% Trunc	Estimate
19	Run Mean	12.4326	—	-20.9436	0	0	0	0.00%	71.85243
	MSEAT	17.1281	0	-18.9146	0	0	14869	7.08%	73.88143
	MSEAET	16.2740	0	-19.5719	0	0	16374	7.80%	73.22410
	MSEASVT	17.1281	0	-18.9146	0	0	14869	7.08%	73.88143
	MSEASVT(A)	61.2140	0	0.1310	1	1	204473	97.37%	92.92698
20	Run Mean	11.0520	—	-11.6713	0	0	0	0.00%	81.12466
	MSEAT	15.4298	0	-12.1024	0	0	93814	44.67%	80.69361
	MSEAET	13.2481	0	-12.2774	0	0	93420	44.49%	80.51861
	MSEASVT	15.3983	0	-12.1138	0	0	93782	44.66%	80.68224
	MSEASVT(A)	43.5039	0	-0.0721	1	1	199533	95.02%	92.72392
21	Run Mean	19.6149	—	-5.8000	1	0	0	0.00%	86.99604
	MSEAT	20.0111	0	-5.5982	1	0	29911	14.24%	87.19783
	MSEAET	16.5615	1	-8.6258	1	0	63148	30.07%	84.17023
	MSEASVT	17.7943	1	-5.0304	1	0	9030	4.30%	87.76559
	MSEASVT(A)	29.9221	0	-0.1243	1	1	146545	69.78%	92.67168
22	Run Mean	10.3334	—	-17.8674	0	0	0	0.00%	74.92856
	MSEAT	16.1384	0	-14.7844	0	0	76373	36.37%	78.01159
	MSEAET	10.3556	0	-15.1333	0	0	55575	26.46%	77.66272
	MSEASVT	16.1384	0	-14.7844	0	0	76373	36.37%	78.01159
	MSEASVT(A)	13.9737	0	0.2904	1	1	209886	99.95%	93.08641
23	Run Mean	21.3943	—	-15.2324	0	0	0	0.00%	77.56365
	MSEAT	22.1491	0	-14.8770	0	0	1121	0.53%	77.91904
	MSEAET	10.3177	1	-22.1375	0	0	39698	18.90%	70.65849
	MSEASVT	22.1474	0	-14.8773	0	0	1119	0.53%	77.91866
	MSEASVT(A)	36.4812	0	-0.0427	1	1	202488	96.42%	92.75326
24	Run Mean	16.6080	—	-10.3719	0	0	0	0.00%	82.42412
	MSEAT	16.2111	1	-9.9460	0	0	1443	0.69%	82.84998
	MSEAET	10.2772	1	-11.9794	0	0	46022	21.92%	80.81656
	MSEASVT	16.2094	1	-9.9468	0	0	1441	0.69%	82.84917
	MSEASVT(A)	40.6451	0	-0.1324	1	1	188427	89.73%	92.66363
25	Run Mean	13.0619	—	0.5479	1	1	0	0.00%	93.34386
	MSEAT	26.3958	0	5.5856	1	0	102351	48.74%	98.38164
	MSEAET	13.8124	0	4.0728	1	1	84370	40.18%	96.86876
	MSEASVT	13.5138	0	2.9391	1	1	9352	4.45%	95.73510
	MSEASVT(A)	17.5576	0	-0.1225	1	1	57957	27.60%	92.67346

Micro Analysis of Results

Model: M / M / 2 & M / M / 3 Tandem Queue w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Estimate
1	Run Mean	1.88088308	—	-0.17701849	1	1	38.8774483
	MSEAT	0.36183731	1	0.08973975	1	1	39.1442066
	MSEAET	0.55970889	1	-0.00041470	1	1	39.0540521
	MSEASVT	0.36070168	1	4.40147513	0	0	43.4559419
2	Run Mean	1.88371299	—	-0.53087886	1	1	38.5235879
	MSEAT	0.11193031	1	-0.06224839	1	1	38.9922184
	MSEAET	0.35107230	1	-0.14909015	1	1	38.9053766
	MSEASVT	0.11391549	1	-0.05815264	1	1	38.9963142
3	Run Mean	2.20751141	—	-0.27642904	1	1	38.7780378
	MSEAT	0.68614878	1	0.14334373	1	1	39.1978105
	MSEAET	0.86605538	1	-0.08822773	1	1	38.9662391
	MSEASVT	0.75234751	1	0.00066596	1	1	39.0551328
4	Run Mean	2.19127572	—	0.23021222	1	1	39.2846790
	MSEAT	0.34113488	1	0.17178055	1	1	39.2262474
	MSEAET	0.54280286	1	-0.14055316	1	1	38.9139136
	MSEASVT	0.33864457	1	0.17700138	1	1	39.2314682
5	Run Mean	1.85571540	—	-0.44205254	1	1	38.6124143
	MSEAT	0.16965721	1	-0.06707873	1	1	38.9873881
	MSEAET	0.49766200	1	-0.36016590	1	1	38.6943009
	MSEASVT	0.17047624	1	-0.06636046	1	1	38.9881063
6	Run Mean	1.57998860	—	0.70257933	1	1	39.7570461
	MSEAT	0.05616385	1	-0.02539351	1	1	39.0290733
	MSEAET	0.33290827	1	-0.30908507	1	1	38.7453817
	MSEASVT	0.13472930	1	0.02198484	1	1	39.0764516
7	Run Mean	2.82011952	—	1.25846776	1	0	40.3129346
	MSEAT	0.55895040	1	-0.16695775	1	1	38.8875090
	MSEAET	0.83483671	1	-0.48617052	1	1	38.5682963
	MSEASVT	0.61694225	1	-0.31465720	1	1	38.7398096
8	Run Mean	2.73433349	—	1.04447331	1	0	40.0989401
	MSEAT	0.04923712	1	-0.02159305	1	1	39.0328738
	MSEAET	0.32143233	1	-0.02416590	1	1	39.0303009
	MSEASVT	0.17444334	1	-0.00371573	1	1	39.0507511
9	Run Mean	1.14055941	—	-0.21569416	1	1	38.8387726
	MSEAT	0.07358185	1	0.03757109	1	1	39.0920379
	MSEAET	0.34419201	1	0.08459012	1	1	39.1390569
	MSEASVT	0.08730124	1	0.03420436	1	1	39.0886712
10	Run Mean	1.98043397	—	0.05419663	1	1	39.1086634
	MSEAT	0.41346971	1	-0.50519761	1	1	38.5492692
	MSEAET	0.77913210	1	-0.84006807	1	0	38.2143987
	MSEASVT	0.40368513	1	-0.53360288	1	1	38.5208639

Micro Analysis of Results

Model: M / M / 2 & M / M / 3 Tandem Queue w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Estimate
11	Run Mean	2.12633665	—	2.33708251	0	0	41.3915493
	MSEAT	0.00072568	1	-0.00004492	1	1	39.0544219
	MSEAET	0.32328446	1	-0.41035351	1	1	38.6441133
	MSEASVT	0.01601172	1	-0.00292768	1	1	39.0515391
12	Run Mean	1.59358140	—	-1.29143581	1	0	37.7630310
	MSEAT	0.31762519	1	0.16603629	1	1	39.2205031
	MSEAET	0.51462227	1	-0.02180223	1	1	39.0326646
	MSEASVT	0.34397725	1	0.15015867	1	1	39.2046255
13	Run Mean	2.97921980	—	2.57353080	0	0	41.6279976
	MSEAT	0.05944471	1	-0.03155990	1	1	39.0229069
	MSEAET	0.11998624	1	0.05241122	1	1	39.1068780
	MSEASVT	0.51304388	1	0.21383316	1	1	39.2683000
14	Run Mean	1.94285236	—	-1.11197451	1	0	37.9424923
	MSEAT	0.57780325	1	-0.44064256	1	1	38.6138242
	MSEAET	0.69073710	1	-0.54705553	1	1	38.5074113
	MSEASVT	0.64002275	1	-0.31130384	1	1	38.7431630
15	Run Mean	1.91281267	—	-0.87298233	1	0	38.1814845
	MSEAT	0.00272298	1	0.00153336	1	1	39.0560002
	MSEAET	0.17783368	1	-0.17405990	1	1	38.8804069
	MSEASVT	0.20404725	1	-0.11909473	1	1	38.9353721
16	Run Mean	1.94673988	—	0.36211518	1	1	39.4165820
	MSEAT	0.02101926	1	0.01003209	1	1	39.0644989
	MSEAET	0.46855165	1	0.01893635	1	1	39.0734032
	MSEASVT	0.02962931	1	0.01288387	1	1	39.0673507
17	Run Mean	1.90741640	—	-0.05560888	1	1	38.9988579
	MSEAT	0.10926739	1	-0.06626683	1	1	38.9882000
	MSEAET	0.37053203	1	-0.41468332	1	1	38.6397835
	MSEASVT	0.10866042	1	-0.06875577	1	1	38.9857110
18	Run Mean	1.14249854	—	-0.59942411	1	1	38.4550427
	MSEAT	0.00112829	1	-0.00027464	1	1	39.0541922
	MSEAET	0.24168178	1	-0.21960609	1	1	38.8348607
	MSEASVT	0.01846578	1	-0.00662219	1	1	39.0478446
19	Run Mean	1.67428523	—	0.62928487	1	1	39.6837517
	MSEAT	0.00174473	1	0.00119870	1	1	39.0556655
	MSEAET	0.22302544	1	-0.14909343	1	1	38.9053734
	MSEASVT	0.00494922	1	-0.00614729	1	1	39.0483195
20	Run Mean	1.87075654	—	0.36409448	1	1	39.4185613
	MSEAT	0.06293412	1	-0.03209983	1	1	39.0223670
	MSEAET	0.33570175	1	-0.06439118	1	1	38.9900756
	MSEASVT	0.06813047	1	-0.03601711	1	1	39.0184497

Micro Analysis of Results

Model: M / M / 2 & M / M / 3 Tandem Queue w / Initial Q Length

Exp#	Est. Method	1/2 Width	Less Y/N	Bias	± 5% θ	± 2% θ	Estimate
21	Run Mean	2.14897503	—	-1.04447159	1	0	38.0099952
	MSEAT	0.64780638	1	-0.44503397	1	1	38.6094328
	MSEAET	0.81041109	1	-0.51433441	1	1	38.5401324
	MSEASVT	0.74655093	1	-0.31116412	1	1	38.7433027
22	Run Mean	2.79861262	—	-0.52364969	1	1	38.5308171
	MSEAT	0.62956796	1	-0.13410550	1	1	38.9203613
	MSEAET	0.93636753	1	-0.35996210	1	1	38.6945047
	MSEASVT	0.62814569	1	-0.15262745	1	1	38.9018393
23	Run Mean	2.45376960	—	0.24170791	1	1	39.2961747
	MSEAT	0.27002224	1	-0.18335984	1	1	38.8711070
	MSEAET	0.58879009	1	-0.54440217	1	1	38.5100646
	MSEASVT	0.28363967	1	-0.15450231	1	1	38.8999645
24	Run Mean	1.47267735	—	-0.68594312	1	1	38.3685237
	MSEAT	0.29455727	1	-0.25856133	1	1	38.7959055
	MSEAET	0.52408700	1	-0.52388516	1	1	38.5305816
	MSEASVT	0.29551514	1	-0.26542012	1	1	38.7890467
25	Run Mean	1.83565182	—	-0.22995311	1	1	38.8245137
	MSEAT	0.15821824	1	-0.12080743	1	1	38.9336594
	MSEAET	0.44615869	1	-0.38342805	1	1	38.6710387
	MSEASVT	0.15830547	1	-0.13301245	1	1	38.9214544

APPENDIX C. GLOSSARY

Absolute Error — A method to measure error associated with the sample mean estimate of the true mean. The absolute error equals the absolute value of the difference between the sample mean and the true mean. This is also known as the absolute value of the bias associated with an estimate.

$$\beta = \left| \bar{X} - \mu \right|$$

Central Limit Thm — Let $Z_n = \left[\bar{X}(n) - \mu \right] / \sqrt{\sigma^2/n}$ and let Fn(z) be the distribution function of Zn for a sample size of "n". The central limit theorem is: If the sample size "n" of observations of a random variable Zn is sufficiently large, then the random variable will be approximately distributed as a standard normal random variable.

Initialization Bias — In simulation output, when the initial conditions are such that they are representative of the steady state conditions, the estimates produced from a finite simulation run length may be biased as a result. This bias of the estimate, $\bar{X} - \theta$, (where $\bar{X}$ is the sample mean and θ is the true values) is referred to as initialization bias.

Law of Large Numbers — If $X_1, \ldots\ldots, X_n$, is a random sample from a distribution with finite mean μ and variance σ^2, then the sequence of sample means converges in probability to μ.

$\bar{X}_n \xrightarrow{P} \mu$.

Mean Square Error — The expected value of the squared difference between the sample mean and the true mean. This definition easily derives an equation using the definitions of bias and the variance of an estimator.

$\mathrm{MSE} = \left\{ \mathrm{Bias}\left[\bar{X}(n) \right] \right\}^2 + \sigma^2\left[\bar{X}(n) \right]$ This is possible since bias and variance equal the following:

$$\mathrm{Bias}\left[\bar{X}(n) \right] = \mathrm{E}\left[\bar{X}(n) - \mu_x \right]$$

$$Var\left[\bar{X}(n) \right] = E\left[\left\{ \bar{X}(n) - E\left[\bar{X}(n) \right] \right\}^2 \right]$$

φ—mixing — A sequence of stationary random variables, (x1,....,xn), is defined on a probability space (W, b, P). Basically, the process of observations of mapped random variables (Y1,.....,Yn) is said to be phi—mixing if Yi and Yi+j become independent as j becomes large. In a phi—mixing process the distant future is virtually independent of the past and present . This property gives strong Markov process similarities and allows for easier asymptotic evaluations.

Random Variable — a real valued function defined on a sample space. That is, if W is an experiment having sample space Z, and F is a function which assigns a real number F(x) to every outcome {x is an element of Z}, then F(x) is called a random variable.

Discrete Random Variable — if the range space of the random variable F(x) is either countably infinite or finite, then F(x) is a discrete random variable. That is, x takes on values 1, 2, 3, ...only.

Continuous Random Variable — If the range space of the random variable F(x) is an interval or a collection of intervals, then F(x) is a continuous random variable. That is, x takes on any value between 1 and 2 for example. There is an associated F(x) value for any possible value in the interval.

Relative Error — A method to measure error associated with the sample mean estimate of the true mean. The relative error equals the absolute value of the difference between the sample mean and the true mean divided by the absolute value of the true mean. $\gamma = \left|\bar{X} - \mu\right| / |\mu|$

Steady State Dist. — For an given value of "x" and set "I" where F(x) is a random variable and "I" is the set initial conditions, if the function of x given the set initial conditions, Fi(x | I), goes to the value of F(x), then F(x) is called the steady state distribution. This is affected by the desired precision.

Stochastic Process — a collection of random variables {X(t), t is an element of T} defined on a common probability space indexed by the index set T (usually T is time) which describes the evolution of the system. The set of all possible values that the random variable X(t) can take is called the state space.

APPENDIX D. REFERENCES

[1] Altiok, Tayfur (July-August 1989), "Approximate Analysis of Queues in Series with Phase-Type Service Times and Blocking", Operations Research, vol. 37, No. 4, pp. 601-610.

[2] Banks, Jerry and John S. Carson (1984), Introduction to Discrete-Event System Simulation, Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

[3] Billingsley, Patrick (1968), Convergence of Probability Measures, New York: John Wiley & Sons, Inc.

[4] Cheng, R. C. H. (1976), "A Note on the Effect of Initial Conditions on A Simulation Run", Operational Research Quarterly", vol. 27, No. 2, pp. 467-470.

[5] Conway, R.W. (1963), "Some Tactical Problems in Digital Simulation", Management Science, vol. 10, No. 1, pp. 47-61.

[6] Fishman, George S. (1972), "Bias Considerations in Simulation Experiments", Operations Research, vol. 20, pp. 785-790.

[7] Gafarian, A.V. C.J. Ancker, Jr. and T Morisaku (1978), "Evaluation of Commonly Used Rules for Detecting 'Steady State' in Computer Simulation", Naval Research Logistics Quarterly, vol. 25, pp. 511-529.

[8] Glaserman, Paul. and David D. Yao (June 1992), "Some Guidelines and Gaurantees for Common Random Numbers", Management Science, vol. 38, No. 6, pp. 884-907.

[9] Glynn, Peter W. and Philip Heidelberger (March 1992), "Experiments with Initial Transient Deletion for Parallel, Replicated Steady-State Simulations", Management Science, vol. 38, No. 3, pp. 400-418.

[10] Glynn, Peter W. and Donald L. Iglehart, "A New Initial Bias Deletion Rule", Proceedings of the 1987 Winter Simulation Conference, pp. 318-319.

[11] Gross, Donald and Carl M. Harris (1985). Fundamentals of Queueing Theory, 2d Edition, New York, New York: John Wiley & Sons.

[12] Hillier, Frederick S. and Gerald J. Lieberman (1986), Introduction to Operations Research, 4th Edition, USA: Holden-Day, Inc.

[13] Hillier, Frederick S. and Oliver S. Yu (1981), Queueing Tables and Graphs, New York: North Holland.

[14] Hines, William H. and Douglas C. Montgomery (1980), Probability and Statistics in Engineering and Management Science, 2d Edition, New York, New York: John Wiley & Sons.

[15] Kelton, W. David and, Averill M. Law, (1983) "Transient Behavior of the M/M/s Queue, with Implications for Steady State Simulation", Operations Research, vol. 30, pp. 641-658.

[16] Kelton, W. David (1985), "Transient Exponential-Erlang Queues and Steady-State Simulation", Communications of the ACM, vol. 28, No. 7, pp. 741-749.

[17] Kelton, W. David (1989) "Random Initialization Methods In Simulation", IIE Trans. vol. 21, pp. 355-367.

[18] Kimura, Toshikazu (1991), "Approximations for Networks of Queues with Overtime", Management Science, vol. 37, No. 3, pp. 282-300.

[19] Kouvatsos, Demetres D. (1983) "A Maximum Entropy Analysis of the GI/G/1 Queue at Equilibrium", Journal of the Operations Research Society, vol. 39, No. 2, pp. 183-200.

[20] Law, Averill M. and W. David Kelton (1991) Simulation Modeling and Analysis, 2d Edition, USA: McGraw-Hill, Inc.

[21] Law, Averill M. and W. David Kelton (1983) "A New Approach for Dealing with the Startup Problem in Discrete Event Simulation", Naval Research Logistics Quarterly, vol. 30, pp. 641-658.

[22] Ma, X. and A. K. Kochhar (1983) "A Comparison Study of Two Tests for Detecting Initialization Bias in Simulation Output", Simulation, vol. 61, 2, pp. 94-101.

[23] Minh, Do Le (September 1987), "Simulating GI/G/k Queues in Heavy Traffic", Management Science, vol. 33, No. 9, pp. 1192-1199.

[24] Nelson, Barry L. (1989), "Variance Reduction in the Presence of Initial-Condition Bias", Working Paper Series, Department of Industrial Systems Engineering, The Ohio State University, pp. 1-36.

[25] Odoni, Amedeo R. and Emily Roth, (May-June 1983) "An Empirical Investigation of the Transient Behavior of Stationary Queueing Systems", Operations Research, vol. 31, No. 3, pp. 432-455.

[26] Pawlikowski, Krzysztof, (Jun 1990), "Steady State Simulation of Queueing Processes: A Survey of Problems and Solutions", AMC Computer Surveys, vol. 22, No. 2, pp. 123-170.

[27] Ravindran, A., Don T. Phillips, and James J. Solberg (1987), Operations Research, Principles and Practice (2d. ed.), New York, New York: John Wiley & Sons.

[28] Ross, Sheldon M. (1990), A Course in Simulation, New York, New York: Macmillan, Inc.

[29] Schruben, Lee W. and H. Singh (November/December 1983), "Optimal Tests for Initialization Bias in Simulation Output", Operations Research, vol. 31, no. 6, pp. 1167-1177.

[30] Schruben, Lee W.(May/June 1982), "Detecting Initialization Bias in Simulation Output", Operations Research, vol. 30, no. 3, pp. 569-590.

[31] Snell, Mark and Lee W. Schruben (1984), "Weighting Simulation Data to Reduce Initialization Effects", IIE Transactions, vol. 17, No. 4, pp. 354-363.

[32] Tijms, Henk C. (1986), Stochastic Modeling and Analysis: A Computational Approach, New York: John Wiley & Sons, Inc.

[33] Tijms, Henk C., L.P. Seelen and M.H. VanHorn (1985), Tables for Multi-Server Queues, North Holland: Elsevier Science Publishers.

[34] Vassilacopolous, G. (1989) "Testing for Initialization Bias in Simulation Output", Simulation, vol. 52, pp. 151-153.

[35] Welch, Peter D. (1983), "A Graphical Approach to the Initial Transient Problem In Steady State Simulations", 10th IMACS World Congress on System Simulation and Scientific Computation, pp. 219-221.

[36] White, K. Preston and Mary A. Minnox, "A Simple Rule for Mitigating Initialization Bias in Simulation Output", To Be Published in Proceedings of the 1994 IEEE Conference on Man, Systems and Cybernetics.

[37] Whitt, Ward, (November 1989), "Planning Queueing Simulations", Management Science, vol. 35, No. 11, pp. 1341-1366

[38] Whitt, Ward, (May 1992), "Understanding the Efficiency of Multi-Server Service Systems", Management Science, vol. 38, No. 5, pp. 708-723.

[39] Whitt, Ward, (Jun 1991), "The Efficiency of One Long Run Versus Independent Replications in Steady State Simulation", Management Science, vol. 37, No. 6, pp. 645-665

[40] Whitt, Ward, (December 1992), "Approximations for the GI/G/m Queue", To Be Published in POM.

[41] Wilson, James R. and Alan B. Pritsker (August 1978) "A Survey of Research on the Simulation Startup Problem", Simulation , pp. 55-58.

[42] Wilson, James R. and Alan B. Pritsker (September 1978) "Evaluation of Startup Policies in Simulation Experiments", Simulation , pp. 79-89.